Inaugural-Dissertation

zur

Erlangung der Doktorwürde

der

Naturwissenschaftlich-Mathematischen

Gesamtfakultät

der

Ruprecht-Karls-Universität

Heidelberg

Vorgelegt von:

Diplom-Mathematiker & Diplom-Physiker

Hinderk Martens Buß

aus Emden

Tag der mündlichen Prüfung:

09.10.2003

Thema

A posteriori Error Estimators based on Duality Techniques from the Calculus of Variations

Gutachter:

1.) Prof. Dr. Rolf Rannacher

2.) Prof. Dr. Hans-Georg Bock

A posteriori Error Estimators based on Duality Techniques from the Calculus of Variations

Hinderk M. Buß
Institute of Applied Mathematics
Im Neuenheimer Feld 293
69120 Heidelberg, Germany

October 15, 2003

Abstract

A theoretical framework is presented within which we can systematically develop a posteriori error estimators for any variational statement of the form

$$F(\Lambda x) + G(x) \longrightarrow \min \quad .$$

We merely have to require, that the linear operator Λ be coercive and that the functional F be uniformly convex. As the convex functional G may be arbitrary, the theory can also cover constrained variational formulations. Two applications are discussed in detail: the Dirichlet Problem and the Obstacle Problem. A number of technical issues is considered as well, which pertain to the evaluation of the proposed error bounds using finite element methods: Inter alia a novel non-conforming discretisation scheme for the dual formulation is analysed. The resulting algebraic problem may be solved by a new preconditioned relaxation method, for which a proof of convergence is supplied.

Zusammenfassung

Ein allgemeiner theoretischer Rahmen wird entworfen, der die systematische Entwicklung von a posteriori Fehlerschätzern für Variationsprobleme der Form

$$F(\Lambda x) + G(x) \longrightarrow \min$$

ermöglicht. Vorausgesetzt wird neben der Koerzivität des linearen Operators Λ lediglich die uniforme Konvexität des Funktionals F. Das Funktional G wird als konvex angenommen, so daß auch restringierte Variationsprobleme betrachtet werden können. Anwendungen der Theorie werden in Gestalt des Dirichlet- bzw. des Hindernis-Problems diskutiert. Praktische Fragen werden erörtert, die mit der Auswertung der Fehlerschätzer im Rahmen einer FEM-Simulation zusammenhängen: u. a. wird eine nicht-konforme Diskretisierungsmethode für die duale Formulierung vorgestellt und ein Konvergenzbeweis für ein neues präkonditioniertes Relaxationsverfahren angegeben, mit dessen Hilfe sich das diskretisierte Problem lösen läßt.

Contents

Introduction

The computation of an *a posteriori* estimate for the approximation error we have incurred in the numerical solution of some variational problem requires a mechanism inherent to the problem which relates the distance between our numerical and the true solution to those quantities, we can immediately derive from the information available to us, namely the data of the problem and the numerical solution itself. Let us assume, that X and Y are two reflexive Banach spaces and that $\Lambda : X \longrightarrow Y$ is some continuous operator. The variational problem, whose solution we want to find, shall be written in the form

$$J(x, \Lambda x) \quad \underset{x \in X}{\longrightarrow} \quad \min \tag{1}$$

whereby $J : X \times Y \longrightarrow \mathbb{R}$ denotes some function. In general, we can neither assert, that there is a minimiser to the above problem, nor that its solution - if it exists - is unique. If the function J is convex, we can at least confirm the existence of a unique minimiser $x_0 \in X$. However, this much structure is still insufficient to warrant the existence of an a posteriori estimate for any numerical approximation $x_h \in X$ to the true solution. What we need is an estimate of the form

$$\phi(\| x - x_0 \|_X) \quad \leq \quad J(x, \Lambda x) \;\; - \;\; J(x_0, \Lambda x_0)$$

with $\phi : \mathbb{R}_0^+ \longrightarrow \mathbb{R}_0^+$ denoting a monotonously increasing function. The existence of such a *forcing function* ϕ can be viewed as the defining quality of an *uniformly convex* functional. We may surmise, therefore, that the uniform convexity of the functional J is an indispensable prerequisite for any type of a posteriori error estimate. If $J_* \in \mathbb{R}$ denotes an arbitrary lower bound to the quantity $J(x_0, \Lambda x_0)$ such an error estimate would read:

$$\| x - x_0 \|_X \quad \leq \quad \phi^{-1}\big(\, J(x, \Lambda x) \; - \; J_* \, \big) \quad .$$

For the sake of conciseness, we will henceforth call an inequality of the above description a *hypercycle estimate*. Though this terminology is perhaps not perfectly adequate (see [118] and sections 4.1.6-8 in [135]), it may be derived from the special case of quadratic forms by an easy generalisation. It is not obvious, how such an estimate should be related to those a posteriori error estimates, that can be found most commonly discussed in the contemporary literature. Before we outline the research that has been conducted in this field, let us work out an example and demonstrate which links exist between a conventional residual based a posteriori error estimator for the Laplace problem and hypercycle estimates for the Dirichlet integral.

Residual based, explicit error estimators

Bluntly put, all a posteriori error estimators, which rely on the computation of certain residual expressions, implicitly assume two requirements to be met by the variational formulation and by the numerical approximation x_h: There must exist an a priori estimate, with the help of which the approximation error can be bounded by some residual expression. To achieve the highest accuracy possible this expression has to be evaluated in a dual norm, however. And secondly, the numerical approximation x_h must feature a best approximation property, which allows for an estimate of that very norm. While the first condition is met qua definitionem by all elliptic linear operators, widely known under the denomination *Cea's lemma*, the best approximation property is usually referred to as *Galerkin orthogonality*. It can only be satisfied in a finite element context, if we assume that all integrals involved are evaluated exactly. To illustrate our point, let us consider some square integrable function $f \in L^2(\Omega)$ defined on some bounded domain $\Omega \subset \mathbb{R}^2$. Our aim is to find a square integrable function with square integrable first derivatives $x_0 \in H_0^1(\Omega)$, which solves the Dirichlet problem

$$- \; \Delta \, x \quad = \quad f \tag{2}$$

with homogeneous boundary conditions in a weak sense. (For a rigorous definition of our notation for the various function spaces and those inner products, we are going to employ, we refer to

section 2.1.1). Let us suppose, that X_h denotes a finite dimensional subspace of $H_0^1(\Omega)$ and that $x_h \in X_h$ represents the solution of the following variational problem:

$$\langle \nabla x_h, \nabla \psi_h \rangle_\Omega \quad = \quad (f, \psi_h)_\Omega \qquad ; \qquad \psi_h \in X_h \quad . \tag{3}$$

Due to the special form of the norm $|\cdot|_{\Omega,1}$, which the space $H_0^1(\Omega)$ is equipped with, the elliptic regularity of the variational formulation leads to an almost trivial a priori estimate:

$$| x_0 - x_h |_{\Omega,1} \qquad \leq \qquad \sup_{\psi \in H_0^1(\Omega)} \frac{(f, \psi)_\Omega - \langle \nabla x_h, \nabla \psi \rangle_\Omega}{| \psi |_{\Omega,1}} \qquad =: \qquad R(x_h) \quad .$$

Since the numerical approximation x_h meets the optimality condition (3), we are able to bound the residual $R(x_h)$ by exploiting a result on the approximation properties of the canonical finite element interpolation operator $\Pi : H_0^1(\Omega) \longrightarrow X_h$:

$$C_M \; > \; 0 \qquad : \qquad \| \psi \, - \, \Pi \psi \|_M \;\; \leq \;\; C_M \;\; h_M \;\; | \psi |_{M,1} \qquad ; \qquad \psi \in H_0^1(\Omega) \;\; .$$

Hereby, the symbol M shall denote a simplicial patch, as it is defined by a decomposition $\mathcal{M}_h$ of the computational domain Ω, and h_M its diameter. We may envisage $\mathcal{M}_h$ as the mesh, which we have obtained from our finite element software or which we have calculated with the help of some dedicated software tool. For simplicity, let us assume the domain Ω to be polyhedral, such that we do not need to discuss contributions to the error estimator, which stem from an inadequate resolution of the boundary $\partial\Omega$. Analytical techniques similar to those discussed in [59] may be employed to obtain bounds on various other functions of the defect. For instance:

$$\hat{C}_M \; > \; 0 \qquad : \qquad \| \psi \, - \, \Pi \psi \|_{\partial E} \;\; \leq \;\; \hat{C}_M \;\; \sqrt{h_E} \;\; | \psi |_{M,1} \qquad ; \qquad \psi \in H_0^1(\Omega)$$

with $E \subset \partial M$ denoting some subset of the element's boundary and h_E designating the length of that curve. If an interior edge E belongs to two elements M_1 and M_2, we may define the jump $[\sigma_h]$ across E of any vector field σ_h, which is sufficiently regular both in M_1 and in M_2, with the help of the outward pointing normal vectors $n_1 \in \mathbb{R}^2$ and $n_2 = -n_1$ perpendicular to E:

$$[\sigma_h](\xi) \quad := \quad n_1 \cdot \sigma_h \big|_{M_1}(\xi) \;\; + \;\; n_2 \cdot \sigma_h \big|_{M_2}(\xi) \qquad ; \qquad \xi \in E \quad .$$

Let us abbreviate the set of all interior edges by $\mathcal{E}_h$. We may choose $\psi_h = \Pi \psi$ as a test function in (3) and perform a partial integration on each patch $M \in \mathcal{M}_h$ of the computational domain. By invoking the above approximation results for the interpolation operator Π and applying Hölder's inequality we can thereupon assert the existence of a stability constant $C > 0$, such that:

$$\begin{aligned} R(x_h) \quad &= \quad \sup_{\psi \in H_0^1(\Omega)} \frac{1}{| \psi |_{\Omega,1}} \sum_{M \in \mathcal{M}_h} \Big\{ \int_M (f + \Delta x_h)(\psi - \psi_h) \;\; - \;\; \frac{1}{2} \oint_{\partial M} (\psi - \psi_h) \Big[\frac{\partial x_h}{\partial n} \Big] \;\; \Big\} \\ &\leq \quad C \;\; \Big\{ \sum_{M \in \mathcal{M}_h} h_M^2 \, \| f + \Delta x_h \|_M^2 \;\; + \;\; \sum_{E \in \mathcal{E}_h} h_E \, \Big\| \Big[\frac{\partial x_h}{\partial n} \Big] \Big\|_E^2 \;\; \Big\}^{1/2} \quad =: \quad P(x_h) \;\; . \end{aligned}$$

The quantity $P(x_h)$ we have thus derived can be viewed as an a posteriori error estimator for the solution x_h of the variational problem (3). Any other choice of the function $x_h' \in X_h$ will result in an estimate $P(x_h')$ that is unreliable.

Hypercycle estimates

A somewhat different type of energy error estimate can be derived from the observation that we may cast the Dirichlet problem (2) into the following abstract setting: Let us assume, the Hilbert space H has been decomposed into the orthogonal sum of two subspaces U and V. If any two elements $u_0, v_0 \in H$ have been fixed, determine the very function $s \in H$ that is contained in the intersection of the affine spaces $U_0 := U + \{u_0\}$ and $V_0 := V + \{v_0\}$. We note, that is common element $s \in U_0 \cap V_0$ simultaneously solves two distinct minimisation problems:

$$\| u - v_0 \|_H^2 \quad \underset{u \in U_0}{\longrightarrow} \quad \min \qquad ; \qquad \| v - u_0 \|_H^2 \quad \underset{v \in V_0}{\longrightarrow} \quad \min \quad . \tag{4}$$

Since $s-u \in U$ and $s-v \in V$ are orthogonal for any choice of $u \in U_0$ and $v \in V_0$, we find:

$$\frac{1}{4}\,\|\,s\,-\,u\,\|_H^2\;+\;\frac{1}{4}\,\|\,s\,-\,v\,\|_H^2 \quad=\quad \Big\|\,s\,-\,\frac{u+v}{2}\,\Big\|_H^2 \quad=\quad \frac{1}{4}\,\|\,u\,-\,v\,\|_H^2 \quad . \tag{5}$$

The right hand side of the equation (5) can be understood as an a posteriori error estimator for both the numerical solutions u and v of the above minimisation problems. According to (5) the true solution s is located someplace on a sphere around the mean value of these solutions with a radius proportional to $\|\,u-v\,\|_H$. Hence, the result is usually referred to as a *hypercycle estimate*. In the case of the Dirichlet problem (2) the Hilbert space H can be identified as the space $L^2(\Omega,\mathbb{R}^2)$ of all square integrable vector fields. The mutually orthogonal subspaces U and V are obtained by a so called Helmholtz decomposition of $L^2(\Omega,\mathbb{R}^2)$:

$$\begin{aligned} U \quad &:= \quad \big\{\;\sigma \in L^2(\Omega,\mathbb{R}^2)\;\big|\;v \in H_0^1(\Omega)\;:\;\sigma=\nabla v\;\big\} \quad , \\ V \quad &:= \quad \big\{\;\sigma \in L^2(\Omega,\mathbb{R}^2)\;\big|\;\langle\,\sigma,\nabla v\,\rangle_\Omega\;=\;0\;;\;v \in H_0^1(\Omega)\;\big\} \quad . \end{aligned}$$

That means, the subspace U consists of all such vector fields, which can be exhibited as the flow of some potential, while the subspace V contains all the solenoidal vector fields. We must fix the elements u_0 and v_0 in such a way, that the common vector field $s \in L^2(\Omega,\mathbb{R}^2)$ represents the flow of the solution $x \in H_0^1(\Omega)$ to problem (2):

$$s\;\in\;U \qquad : \qquad -\,\nabla\cdot s \quad=\quad f \qquad .$$

Accordingly, we must impose the conditions $u_0=0$ and $\nabla\cdot v_0=-f$. While the first condition can be met immediately, fulfilling the second requirement poses a major difficulty, we have to overcome before we can bound the approximation error. Let us assume, we have found a suitable field v_0 and succeeded in constructing a finite dimensional subspace $V_h \subset V$. By minimising the norm of $\sigma_h \in V_h+\{v_0\}$ we may find an approximation to the gradient ∇x, such that

$$|\,x_0-x_h\,|_{\Omega,1}^2 \qquad\leq\qquad \|\,\nabla x_h\,-\,\sigma_h\,\|_\Omega^2 \qquad =:\qquad 2\,M(x_h,\sigma_h) \tag{6}$$

yields an error estimate, that is as sharp as our choice of the ansatz V_h will permit it.

Error bounds derived from complementary energy functionals

The limitations of conventional hypercycle estimates are obvious: They can only be applied to linear problems posed in a Hilbert space setting and, perhaps more importantly, they require an a priori knowledge of certain elements, with the help of which the orthogonal subspaces are constructed. Against our extending the hypercycle concept to the more general class of minimisation tasks involving uniformly convex functionals let us exhibit the variational problems (4) as dual to one another. The solution of (2) is the minimiser of the functional

$$J(x) \quad:=\quad \frac{1}{2}\,\langle\,\nabla x,\nabla x\,\rangle_\Omega\;-\;(\,f,x\,)_\Omega \qquad ; \qquad x \in H_0^1(\Omega) \quad .$$

To introduce a saddle point formulation let us rewrite this functional in the following form:

$$J(x) \quad=\quad \sup_{\sigma\in L^2(\Omega,\mathbb{R}^2)}\Big\{\;\langle\,\nabla x,\sigma\,\rangle_\Omega\;-\;\frac{1}{2}\,\langle\,\sigma,\sigma\,\rangle_\Omega\;\Big\}\;-\;(\,f,x\,)_\Omega$$

for any $x \in H_0^1(\Omega)$. Hence, the complementary functional $J^*: L^2(\Omega,\mathbb{R}^2) \longrightarrow \mathbb{R}$ is defined by:

$$J^*(\sigma)\;:=\;\inf_{x\in H_0^1(\Omega)}\Big\{\,\langle\,\nabla x,\sigma\,\rangle_\Omega-(\,f,x\,)_\Omega\,\Big\}\;-\;\frac{1}{2}\,\langle\,\sigma,\sigma\,\rangle_\Omega \quad=\quad -\,\chi_{V_0}(\sigma)\;-\;\frac{1}{2}\,\langle\,\sigma,\sigma\,\rangle_\Omega$$

with $\chi_{V_0}: L^2(\Omega,\mathbb{R}^2) \longrightarrow \mathbb{R}$ denoting the *indicator function* of the affine space V_0. By definition of the space $U_0=U$ we can reformulate the first minimisation problem in the following manner:

$$\frac{1}{2}\,\|\,\nabla\xi-v_0\,\|_\Omega^2 \quad=\quad J(\xi)\;+\;\frac{1}{2}\,\langle\,v_0,v_0\,\rangle_\Omega \qquad \underset{\xi\in H_0^1(\Omega)}{\longrightarrow} \qquad \min \quad .$$

If we ignore the constant offset, the task of approximating the gradient ∇x is therefore equivalent to minimising the functional J. The second statement in (4) is obviously but a paraphrase for maximising the functional J^*. Our findings suggests, that we may express the error majorant $M(x_h, \sigma)$, we have introduced in (6), in terms of the complementary energy functionals J and J^*. In fact, a simple computation demonstrates:

$$\frac{1}{2} \, | \, x_0 - x_h \, |^2_{\Omega,1} \quad \leq \quad M(x_h, \sigma) \quad = \quad J(x_h) \; - \; J^*(\sigma) \qquad ; \qquad \sigma \in V_0 \;\; .$$

By defining the forcing function $\phi(t) = 0.5\, t^2$ we can thus return to the abstract setting which has served as a starting point for our deliberations. The lower bound J_* to the infimum of the energy functional is thereby provided by the complementary energy functional J^*, since we may state for any admissible vector field $\sigma \in V_0$ and any approximation $x \in H^1_0(\Omega)$:

$$-\infty \quad < \quad J^*(\sigma) \quad \leq \quad J(x_0) \quad \leq \quad J(x) \qquad .$$

Relaxing the admissibility constraints

Though the concept of duality based error majorants is very general it suffers from the same drawback conventional hypercycle estimates are fraught with: They will not always attain a finite value, but require certain admissibility constraints to be met by the dual variable σ. In the above example, the error estimates have been formulated under the assumption, that $\nabla \cdot \sigma = -f$ be fulfilled. Whether explicitly or implicitly, the dual variable itself is usually constructed as an element of some finite dimensional function space. Hence, meeting the admissibility constraints in full will be impossible in most cases.

One possible solution to this difficulty consists in changing the data of problem, such that the resulting admissibility constraint can be met exactly. The consistency error, we introduce thereby, must be controlled by an a priori error estimate. To illustrate the approach let us consider the Dirichlet problem. If we replace the right hand side $f \in L^2(\Omega)$ by its L^2-projection $P_0 \, f$ onto the space of all piecewise constant functions, defined on some simplicial decomposition $\mathcal{M}_h$ of the computational domain Ω, we can use for instance Raviart-Thomas elements [122] of lowest order to construct a vector field $\varsigma_h \in R^0(\mathcal{M}_h)$ satisfying $\nabla \cdot \varsigma_h = -P_0 \, f$. Let us denote the analytical solution of the Dirichlet problem $-\Delta \, x = P_0 \, f$ by $\tilde{x} \in H^1_0(\Omega)$. Cea's lemma asserts:

$$\big| \, x_0 - \tilde{x} \, \big|_{\Omega,1} \quad \leq \quad C \sup_{\psi \in H^1_0(\Omega)} \frac{(\, f - P_0 \, f, \psi \,)_\Omega}{| \, \psi \, |_{\Omega,1}} \quad = \quad C \sup_{\psi \in H^1_0(\Omega)} \frac{(\, f + \nabla \cdot \varsigma_h, \psi - P_0 \, \psi \,)_\Omega}{| \, \psi \, |_{\Omega,1}} \quad .$$

We may control the right hand side of this equation by invoking an interpolation result for the L^2-projection and applying Hölder's inequality afterwards. Bounding the approximation error $| \, x_h - \tilde{x} \, |_{\Omega,1}$ with a conventional hypercycle estimate we thus obtain the following error estimator:

$$\big| \, x_0 - x_h \, \big|_{\Omega,1} \quad \leq \quad \big\| \, \nabla x_h - \sigma_h \, \big\|_\Omega \quad + \quad \tilde{C} \, \Big\{ \sum_{M \in \mathcal{M}_h} h_M^2 \, \big\| \, f + \nabla \cdot \sigma_h \, \big\|^2_M \, \Big\}^{1/2} \quad , \qquad (7)$$

which is valid for any vector field σ_h from the affine space $V \cap R^0(\mathcal{M}_h) + \{\varsigma_h\}$. As this parameter must be admissible, the computation of a field σ_h, which yields a preferably sharp error estimate, can require substantial effort. One possibility to obtain an acceptable field consists in solving the Dirichlet problem (2) with the help of a dual mixed discretisation scheme:

$$\begin{aligned} \langle \, \tau_h, \sigma_h \, \rangle_\Omega \quad + \quad (\, u_h, \nabla \cdot \tau_h \,)_\Omega \quad &= \quad 0 \qquad\qquad ; \qquad \tau_h \in R^0(\mathcal{M}_h) \;\; , \\ (\, y_h, \nabla \cdot \sigma_h \,)_\Omega \quad &= \quad -\, (f, y_h)_\Omega \qquad ; \qquad y_h \in P^0(\mathcal{M}_h) \;\; . \end{aligned}$$

In this context, the piecewise constant function $u_h \in P^0(\mathcal{M}_h)$ serves as a Lagrange multiplier for the admissibility constraint $\nabla \cdot \sigma_h = -P_0 \, f$. For details we refer to [122].

On the equilibrated residual method

In the previous paragraph a dual mixed discretisation scheme has been proposed to find a vector field which meets a modified admissibility constraint. From a practical point of view, however, the

necessity of performing computations of a higher numerical complexity than those, the original variational problem has required, "merely" to assess the quality of the numerical approximation is detrimental to an error estimator. Hence, hypercycle estimates are usually not applied in the very way we have outlined above. In fact, we can determine a vector field $\varsigma_h \in R^k(\mathcal{M}_h)$ with the property $\nabla\cdot\varsigma_h = -P_k f$ by local calculations only. For any interior edge $E \in \mathcal{E}_h$ of the triangulation let us specify a function $g_E : E \longrightarrow \mathbb{R}$ we suppose to approximate the normal flux of the true solution x_0 to problem (2) across the interface E. Furthermore, let us introduce a unit vector $n_E \in \mathbb{R}^2$ perpendicular to the interface. For any element $M \in \mathcal{M}_h$ of the mesh we may now introduce a local minimisation problem:

$$\| \sigma_M - \nabla x_h \|^2_M \quad \underset{\sigma_h \in Q_M}{\longrightarrow} \quad \min \tag{8}$$

with the set Q_M of admissible vector fields being defined by:

$$Q_M := \left\{ \sigma_M \in R^k(\mathcal{M}_h) \;\middle|\; \nabla\cdot\sigma_M = -P_k f \;\wedge\; n_E\cdot\sigma_M\big|_E = P_k g_E \;;\; E \subset \partial M \right\} .$$

It is easily seen, that a vector field σ is admissible, if its restriction $\sigma|_T$ to any element $M \in \mathcal{M}_h$ is contained in the set Q_M. The difficulty lies elsewhere: We must warrant that none of the sets Q_M is empty. Furthermore, we must ensure that the resulting field ς_h, which we define in an element by element fashion $\varsigma_h|_M := \sigma_M$, does yield a sharp estimate. To achieve both goals simultaneously we must construct the flux functions g_E with care. Usually, the process of finding these functions is referred to as *equilibration*. Let $v_h \in \mathbb{P}_l$ denote a polynomial test function of Lagrange type with a degree l less or equal to k, the order of the Raviart-Thomas ansatz for the dual variable. Performing a partial integration on the element M we find:

$$\int_M v_h f \; d\xi \quad = \quad \int_M \nabla v_h \nabla x_0 \; d\xi \quad - \quad \oint_{\partial M} v_h \frac{\partial x_0}{\partial n} \; ds \quad .$$

Since the dual variable σ_M is intended to approximate the vector field ∇x_0 we can use the above equation to specify the equilibrated fluxes across the element edges. To this end we replace the unknown gradient by its numerical approximation:

$$\int_M \left\{ \nabla v_h \nabla x_h - v_h f \right\} d\xi \quad =: \quad \sum_{E\subset\partial M} n_E^{(M)}\cdot n_E \int_E v_h g_E \; ds \quad . \tag{9}$$

Hereby, $n_E^{(M)} \in \mathbb{R}^2$ denotes the outward directed normal vector of the element M perpendicular to its edge E. As $v_h \in \mathbb{P}_l$ can be a constant function, this definition of the numerical fluxes g_E ensures, that the data of the local Neumann problem is consistent. However, the flux functions are not uniquely defined by (9). Let us suppose, that $E \in \mathcal{E}_h$ is an interior edge and that M_1 and M_2 are its two adjoining elements. We introduce a mean value for the numerical flux by:

$$\left\langle \frac{\partial x_h}{\partial n} \right\rangle_E \quad := \quad \frac{1}{2} \left\{ \frac{\partial x_h}{\partial n}\Big|_{M_1} - \frac{\partial x_h}{\partial n}\Big|_{M_2} \right\} n_E^{(M_1)}\cdot n_E \quad .$$

The scalar product $n_E^{(M_1)}\cdot n_E$ causes the mean value to be invariant under our changing the roles of M_1 and M_2. An ansatz for the flux functions must provide enough flexibility to warrant the solvability of (9) for any choice of the test function $v_h \in \mathbb{P}_l$. For the simplest case $l = 0$ the following ansatz has been proposed [98]:

$$g_E \quad := \quad \left\langle \frac{\partial x_h}{\partial n} \right\rangle_E + \lambda_E$$

with $\lambda_E : E \longrightarrow \mathbb{R}$ denoting a linear function to be determined with a view to (9). Alternatively, a method known as *flux splitting* may be applied [4], which relies on the ansatz:

$$g_E \quad := \quad \left\langle \frac{\partial x_h}{\partial n} \right\rangle_E + \alpha_E \left[\frac{\partial x_h}{\partial n} \right]_E \quad .$$

Hereby, the function $\alpha_E : E \longrightarrow \mathbb{R}$ is again assumed to be linear. Let the index i denote some interior vertex of the mesh and $\psi_i \in P^1(\mathcal{M}_h)$ a continuous function which vanishes at all vertices

except for the vertex i where it attains the value 1. We introduce the set $\mathfrak{M}_i$ of all those elements, which are contained in the support of ψ_i, and define basis functions $L^i_{\scriptscriptstyle E} : E \longrightarrow \mathbb{R}$, which are orthonormal to the traces $\psi_i \,|_{\scriptscriptstyle F}$ if integrated along some edge F. The corresponding linear factors $\lambda^i_{\scriptscriptstyle E} \in \mathbb{R}$ which determine the functions $\lambda_{\scriptscriptstyle E}$ can be obtained from the following linear systems:

$$\sum_{E \subset \partial M} n^{(M)}_{\scriptscriptstyle E} \cdot n_{\scriptscriptstyle E}\, \lambda^i_{\scriptscriptstyle E} \;=\; \langle\, \nabla\psi_i, \nabla x_h \,\rangle_{\scriptscriptstyle M} - (\,\psi_i, f\,)_{\scriptscriptstyle M} - \sum_{E \subset \partial M} n^{(M)}_{\scriptscriptstyle E} \cdot n_{\scriptscriptstyle E} \int\limits_{\scriptscriptstyle E} \psi_i \left\langle \frac{\partial x_h}{\partial n} \right\rangle d\,s \;\;;\;\; M \in \mathfrak{M}_i \;.$$

If the flux splitting scheme is employed, similar equations can be derived which determine the functions $\alpha_{\scriptscriptstyle E}$. The solvability of the above equilibration conditions is discussed for instance in [4]. The resulting a posteriori error estimate is basically (7) with the field σ_h replaced by the field ς_h which we have found by solving the local minimisation problems (8).

Penalising the admissibility constraints

Changing the data of the variational formulation may help to obtain a reliable error bound with the help of a conventional hypercycle estimate. In consequence, the approach is limited to linear problems. Moreover, it suffers from the necessity either to solve a globally defined dual problem or to introduce further approximation errors by specifying numerical fluxes across the element interfaces. If the first approach is used, the computational complexity of evaluating the error estimator is at least comparable to minimising the primal energy functional. If flux conditions are imposed to decouple to the dual formulation into locally defined minimisation problems, the impact of the equilibration procedure on the accuracy of the error estimate must be accounted for. To avoid such drawbacks a different treatment of the admissibility constraints is necessary. In the following we will outline a penalisation of the dual formulation which leads to a duality based a posterior error estimator, that is well defined for any choice of the dual variable. We will assume, that our energy functional has the form

$$J(x, \Lambda x) \;\;:=\;\; F(\Lambda x) \;+\; G(x) \qquad ; \qquad x \in X \tag{10}$$

whereby $G : X \longrightarrow \mathbb{R}$ is a convex and $F : Y \longrightarrow \mathbb{R}$ is an uniformly convex functional. The dual formulation of the problem (1) is specified in terms of the conjugate energy functional

$$J^*(y^*, \Lambda^* y^*) \;\;:=\;\; -\,G^*(-\Lambda^* y^*) \;-\; F^*(y^*) \qquad ; \qquad y^* \in Y^*$$

which is itself defined in terms of the conjugate functionals $G^* : X^* \longrightarrow \mathbb{R}$ and $F^* : Y^* \longrightarrow \mathbb{R}$ in the sense of *Fenchel* [67]. These functionals are themselves convex. Moreover, the functional F^* features a certain amount of smoothness which we will exploit. Since the conjugate functional J^* provides a lower bound for the energy, the abstract error estimate has the following form:

$$\phi(\,\|\, x - x_0 \,\|_{\scriptscriptstyle X}\,) \;\;\leq\;\; M_F(\Lambda x, y^*) \;+\; M_G(x, -\Lambda^* y^*) \qquad ; \qquad y^* \in Y^* \quad . \tag{11}$$

Hereby, the majorant has been split into two contributions which can be attributed to either of the constitutive functionals F and G. The first contribution reads for example:

$$M_F(\Lambda x, y^*) \;\;=\;\; F(\Lambda x) \;+\; F^*(y^*) \;-\; \langle\, \Lambda x, y^* \,\rangle_{\scriptscriptstyle Y} \quad .$$

The second contribution features an analogous structure. We conclude, that the optimal error bound can be expressed with the help of certain *subdifferentials*. (For a short introduction into this subject matter we refer to section 1.1.5.) The first order optimality conditions read:

$$y^* \;\in\; \partial F(\Lambda x) \qquad \wedge \qquad -\Lambda^* y^* \;\in\; \partial G(x) \quad .$$

Hence, the admissibility constraint may be expressed as: $-\Lambda^* y^* \in dom\, G^*$. Hereby, the symbol $dom\, G^*$ denotes the *effective domain* of the conjugate functional G^*, which is basically the very region where G^* attains finite values. Modifying the functional G^* is an option which has been considered briefly in the previous paragraph. Modifying the functional F is a second option, we will explore in the chapter 1. Since we cannot present a detailed account of the procedure in this

introduction, let us merely hint a the underlying idea: The uniform convexity of F implies the existence of an expansion for the Fenchel conjugate, which admits an estimate of the form

$$F^*(z^*) \;\leq\; F^*(y^*) \;+\; \langle\, dF^*(y^*), z^* - y^* \,\rangle_Y \;+\; r^*(\,\|\, z^* - y^* \,\|_{Y^*}\,) \quad ; \quad z^* \in Y^* \;.$$

The function $r : \mathbb{R}_0^+ \longrightarrow [0, +\infty]$ is lower-semicontinuous and convex. Its effective domain is not empty and $r(t) = 0$ implies $t = 0$. A nonnegative functional $R : Y \longrightarrow \mathbb{R}$ with the property $R(0) = 0$ is introduced. With the help of R a second auxiliary functional

$$K_R(s) \quad := \quad \inf_{y \in Y} \left\{\, R^{**}(\Lambda s - y) \;+\; r(\,\|y\|_Y\,) \,\right\} \qquad ; \qquad s \in X$$

may be defined. Any functional $K : X \longrightarrow \mathbb{R}$ that provides a lower bound on K_R can be used to formulate a computable upper bound on the approximation error. Under moderately restrictive assumption on the properties of the linear operator Λ this error bound will stay finite for any admissible approximation $x \in dom G$ and any choice of the dual variable $y^* \in Y^*$:

$$\phi(\,\|\, x - x_0 \,\|_X\,) \;\leq\; M_F(\Lambda x, y^*) \;+\; R(\,\Lambda x - dF^*(y^*)\,) \;+\; \inf_{x^* \in \partial G(x)} K^*(\Lambda^* y^* + x^*) \;. \qquad (12)$$

An application to quadratic forms

The a posteriori error majorant (12) features two contributions, which correspond to the necessary and sufficient optimality conditions for the estimate (11). If the functional $F : H \longrightarrow \mathbb{R}$ defines a positive definite quadratic form on some Hilbert space H, we may choose for instance $R = \kappa\, F$ with $\kappa > 0$ denoting a relaxation parameter. In this special case the energy norm of the residuals both in the first and in the second duality relation is recovered. In fact, the Gâteaux-derivative $dF^*(y^*)$ of the conjugate functional F^* at some point $y^* \in H$ can be identified with the image of y^* under the action of some self-adjoint isomorphism $\Gamma^* : H \longrightarrow H$. We note:

$$F^*(y^*) \quad = \quad \frac{1}{2}\, \langle\, \Gamma^*\, y^*,\, y^* \,\rangle_H \qquad ; \qquad y^* \in H \quad .$$

The growth of F^* may be controlled with the help of the function $r(t) = 0.5\, t^2$, if we define a new norm on H by $\|\, y^* \|^2 := 2\, F^*(y^*)$. Consequently, the auxiliary functional K_R reads:

$$K_R(s) \quad = \quad \inf_{y \in H} \left\{\, \kappa\, F(\Lambda s - y) \;+\; F(y) \,\right\} \quad = \quad \frac{\kappa}{1 + \kappa}\; F(\Lambda s) \qquad ; \qquad s \in X \quad .$$

If the above functional can be used to define a norm on the space X thanks to an inequality of Poincaré-Friedrich type, we may view the second contribution to the error majorant (12) as a residual, which is measured in a dual energy norm defined by:

$$|\, s^* \,|_* \quad := \quad \sup_{s \in X} \frac{\langle\, s, s^* \,\rangle_X}{\sqrt{2\, F(\Lambda s)}} \qquad ; \qquad s^* \in X^* \quad . \qquad (13)$$

Under the assumption that (13) acts as a norm, we may choose $K = K_R$ and thus obtain:

$$K^*(s^*) \quad = \quad \sup_{\lambda \geq 0} \left\{\, \lambda\, |\, s^* |_* \;-\; \frac{\kappa\, \lambda^2}{2\,(1 + \kappa)} \,\right\} \quad = \quad \frac{\kappa + 1}{2\, \kappa}\; |\, s^* \,|_*^2 \qquad ; \qquad s^* \in X^* \quad .$$

A simple computation shows that the first contribution $M_F(\Lambda x, y^*)$ to the error majorant (12) can be expressed in terms of the functional F and the operator Γ^*. The forcing function is given by $\phi(t) = 1/2\, t^2$. Hence, the final a posteriori error estimate reads for any $y^* \in H$:

$$F(\,\Lambda(x - x_0)\,) \quad \leq \quad (1 + \kappa)\; F(\,\Lambda x \;-\; \Gamma^*\, y^*\,) \;+\; \frac{\kappa + 1}{2\, \kappa} \inf_{x^* \in \partial G(x)} |\, \Lambda^* y^* + x^* \,|_*^2 \;.$$

On the relationship between the various error estimators

To illustrate the above result let us apply the calculus to the Dirichlet problem (2). In this case the operator Λ represents the gradient mapping $\nabla : H^1_0(\Omega) \longrightarrow L^2(\Omega, \mathbb{R}^2)$. Thanks to the homogeneous boundary conditions the dual norm (13) is well defined and coincides with the usual norm of the dual space $H^{-1}(\Omega)$, as the functional F is defined by $2\,F(y) = \|y\|^2_\Omega$. The operator $\Gamma^* : L^2(\Omega, \mathbb{R}^2) \longrightarrow L^2(\Omega, \mathbb{R}^2)$ is simply the identity mapping. The subdifferential $\partial G(x)$ contains but the element $-f$ for any function $x \in H^1_0(\Omega)$. After trivial scaling we therefore find the following a posteriori estimate valid for any choice of the field $\sigma \in L^2(\Omega, \mathbb{R}^2)$:

$$|\, x_h - x_0 \,|^2_{\Omega,1} \;\leq\; (1+\kappa)\, \|\, \nabla x_h \,-\, \sigma \,\|^2_\Omega \;+\; (1+\kappa^{-1})\, |\, \nabla\cdot \sigma + f \,|^2_{\Omega,-1} \;=:\; \mathcal{H}^{(\kappa)}(x_h, \sigma) \;\; .$$

If we fix the vector field $\sigma = \nabla x_h$, the first part of the error bound vanishes. We can consider the limit $\kappa \to \infty$ and thus recover the a priori estimate, which the residual based a posteriori estimate $P(x_h)$ is based on. Taking the square root we infer:

$$|\, x_h - x_0 \,|_{\Omega,1} \quad\leq\quad \lim_{\kappa\to\infty} \sqrt{\mathcal{H}^{(\kappa)}(x_h, \nabla x_h)} \quad=\quad R(x_h) \;\; .$$

As a matter of course, the generalised hypercycle estimate $\mathcal{H}^{(\kappa)}$ is not computable. We need to employ the same devices we must bring to bear on conventional explicit error estimators in order to obtain an upper bound we can actually evaluate. Nevertheless, the example indicates, that residual based error estimators may generally be subsumed in our more general framework.

It is natural to look for such vector fields, which turn the quantity $\mathcal{H}^{(\kappa)}(x_h, \sigma)$ into a readily computable error bound. If we discovered a vector field $\varsigma \in H_{\mathrm{div}}(\Omega)$ with the property $\nabla\cdot\varsigma = -f$, we could for instance construct an affine space V_0 by requiring:

$$V_0 \quad := \quad \{\, \varsigma \,\} \;+\; \left\{\; \tau_h \in R^k(\mathcal{M}_h) \;\middle|\; \nabla\cdot\tau_h \,=\, 0 \;\right\} \;\; .$$

Minimising the generalised hypercycle estimate with respect to $\sigma_h \in V_0$ we would be able to suppress the second contribution to $\mathcal{H}^{(\kappa)}(x_h, \sigma_h)$ and fix the equilibration parameter $\kappa = 0$. The resulting error bound would correspond to the conventional hypercycle estimate (6):

$$|\, x_h - x_0 \,|^2_{\Omega,1} \quad\leq\quad \inf_{\sigma_h\in V_0} \mathcal{H}^{(0)}(x_h, \sigma_h) \quad=\quad \inf_{\sigma_h\in V_0} \|\, \nabla x_h \,-\, \sigma_h \,\|^2_\Omega \;\; .$$

Unfortunately, the pivotal field ς is usually not available to us. Hence, the above estimate is rather academic. Relaxing the admissibility constraint we obtain the error bound (7), which consists of an hypercycle estimate and an additional consistency error. Let us define:

$$V_h \quad := \quad \left\{\; \tau_h \in R^k(\mathcal{M}_h) \;\middle|\; \nabla\cdot\tau_h \,=\, -P_k\, f \;\right\} \;\; ,$$

that is, the set of all admissible vector fields. By exploiting the so called Galerkin orthogonality of the function $f - P_k\, f$ and the space $P^0(\mathcal{M}_h)$ we obtain the following upper bound:

$$\mathcal{H}^{(\kappa)}(x_h, \sigma_h) \quad\leq\quad (1+\kappa)\, \|\, \nabla x_h \,-\, \sigma_h \,\|^2_\Omega \;+\; (1+\kappa^{-1})\, C^2\, h^2 \,\|\, f - P_k\, f \,\|^2_\Omega \;\; ,$$

which is valid for all $\sigma_h \in V_h$. Minimising the right hand side of the above inequality with respect to the equilibration parameter κ yields a result analogous to (7):

$$|\, x_h - x_0 \,|_{\Omega,1} \quad\leq\quad \inf_{\kappa>0} \sqrt{\mathcal{H}^{(\kappa)}(x_h, \sigma_h)} \quad\leq\quad \|\, \nabla x_h \,-\, \sigma_h \,\|_\Omega \;+\; C\, h \,\|\, f - P_k\, f \,\|_\Omega \;\; .$$

Hereby, $\sigma_h \in V_h$ is an arbitrary field and $C > 0$ denotes an interpolation constant. Consequently, we may claim our approach to also encompass any conventional hypercycle estimate. Since the equilibrated residual method can be considered a device of obtaining suitable vector fields for an hypercycle estimate pertaining to a variational statement with relaxed admissibility constraints, this particular technique is subsumable as well.

General remarks on a posteriori error computations

To give a complete account of all the results, that have been obtained throughout, roughly speaking, the last 30 years in the field of a posteriori error estimation, is nowadays nigh impossible. Querying for instance the database of the *European Mathematical Society* for publications, which carry the catchphrase *posteriori* in their title, will return more than 700 relevant entries. A survey of the most important techniques and their underlying mathematics can be found in [5, 61, 138].

Broadly speaking, the existing a posteriori error estimators belong to one of three categories: They either attempt to bound the residual of the finite element approximation in some dual norm or they rely on an extrapolation principle. The third category of error estimators employs a complementary formulation of the original problem to find upper and lower bounds on the approximation error. The classical explicit error estimators, which have been introduced by I. Babuška and W. Rheinboldt [14, 15], can be counted in the first category, while the so called *patch recovery* method suggested by O. Zienkiewicz and J. Zhu [142] belongs to the second. It is remarkable, that methods from the third category may claim progeniture: some were discussed as early as 1964 in the papers of S. Mikhlin [109] and B. Fraeijs de Veubeke [71].

The mathematical foundations of those error estimators, which fall into the first respectively third category, have been outlined above. In the following we will not consider any methods from the second category for basically two reasons: these error estimators aim at constructing an approximation to the gradient of the analytical solution, which is more accurate than the gradient of the numerical solution. Hence, we may consider them as hypercycle estimates with the consistency error suppressed as a perturbation of higher order. Secondly, these estimators rely heavily on super-approximation properties to be exhibited by the discretisation procedure. (Concise surveys of the research pertaining to such phenomena can be found in [95, 96].) For all practical purposes, however, the requirements are oftentimes difficult if not even impossible to meet. Therefore, the analysis of the consistency error is most likely to fail and the reliability of the error estimates stays dubious. Gradient recovery schemes, which work on unstructured meshes have been investigated but recently [19, 44, 83].

Those methods used to obtain a posteriori estimates for the energy error in terms of certain residual expressions can also be employed to compute bounds on various other quantities. The methodology advocated in the papers of Rannacher et al. (see [21] for an overview) is based on a technical device attributed to Aubin [10] and Nitsche [113]: The best approximation property of the numerical solution is applied to both arguments of a bilinear form, which has been introduced via a suitably defined *adjoint* problem. The resulting a posteriori estimates contain weights, which may be assessed either by processing the numerical solution to the adjoint problem or by invoking apposite stability results. The latter approach was developed by Babuška and Miller [11–13] and has been taken up among others by Eriksson and Johnson [62, 63]. An alternative approach to the computation of bounds on functional outputs has been promulgated by Patera et al. [104, 115, 116]. Their technique is based on the formulation of a suitably defined saddle point problem and may therefore be counted into the third category of a posteriori estimators.

A posteriori energy error estimates for nonlinear problems have been considered by a number of authors. To give but a few references we note, that [117, 136] contains the analysis of explicit residual estimators, while in [47] an implicit estimator for the energy error is discussed. In [7] a posteriori bounds on the approximation error of a dual mixed Galerkin scheme are derived. The main analytical tool in these papers is essentially the linearisation of the problem combined with appropriate assumptions on the Fréchet derivative of the functional to be minimised. The use of duality techniques from the calculus of variations has been discussed in a paper [126] by S. Repin and Xanthis with a view to plastomechanical problems. The first of these authors has ever since [123–125] elaborated a more general theory for a posteriori error computations based on the relaxation of the dual formulation. Still, his results seem geared towards the special case, that the functional $G : X \longrightarrow \mathbb{R}$ in the context of (10) is linear, and are based on a number of ad hoc assumptions, which seem to limit their applicability even further.

A posteriori estimates for variational problems involving constraints can be classified in the same manner as those estimators for unconstrained problems. Although the literature on such problems is considerable less extensive, we can still find examples for each type of approach: In [93] an estimator based on extrapolation principles is proposed, while an implicit estimator is derived in [6], which relies on the hybridisation of a primal discretisation scheme and may be counted

into the third category. Explicit estimators for the obstacle problem have been investigated in papers [45, 73, 86, 103] authored or coauthored by Johnson, Nochetto and Liu. The former two authors consider a penalty formulation of the state constraints and can thus exploit the best approximation property of the numerical solution. The latter introduces a special projection scheme, by which an admissible approximation is found to the analytical solution of the variational inequality. The resulting error bound consists of two parts: One looks like a conventional a posteriori estimator for the unconstrained problem, while the other accounts for an interpolation error incurred in the first stage of the projection scheme. This second contribution is allegedly a higher order perturbation and hence dropped. In [23] the constraints are practically ignored and a conventional weighted error estimator is defined outside the contact set of the numerical solution. The extrapolation approach suffers from the limited regularity of the analytical solution near the boundary of the contact set (see e. g. [89, 101]). The implicit estimator is dependent on the proper statement of the local subproblems, while the other approaches are critically affected either by the penalisation or the proper resolution of the contact set. Duality techniques from the calculus of variations have already been considered by Hlaváček et al. [82], who applied them to unilateral boundary value problems. A posteriori error estimates for the obstacle problem have been derived in [41, 42] by introducing the state constraint into the saddle point formulation of the primal problem with the help of an additional Lagrange multiplier.

On the contents of this Dissertation

Along the lines of our above exposition we shall develop a more general approach to a posteriori error computations for the numerical solutions of certain variational problems. Our deliberations form the fairer part of chapter 1 and are intended to lead in easy stages to an abstract bound on the approximation error, which is measured in the energy norm. We conclude the first chapter with a discussion of our possibilities to extend our theoretical framework to the computation of bounds on functional outputs, which depend on the solution of the variational problem.

In chapter 2 two generic applications will be considered: the Laplace and the so called Obstacle problem. The latter serves as a simple example for a variational inequality. In both cases, we consider the impact of penalising the admissibility constraint and discuss the accuracy of the resulting hypercycle estimates. In the case of the obstacle problem we compare our findings to those results we have obtained by the alternative approach mentioned above. Using duality techniques we are able to exhibit the generalised hypercycle estimates as complementary energy functionals, which are associated with a differential operator of Helmholtz type. Thus the very nature of the penalisation procedure becomes apparent.

The next chapter deals with those issues, which arise out of the necessity to find the dual parameters present in the hypercycle estimates in some finite dimensional trial space. The bulk of the chapter is dedicated to the discussion of a novel hybridisation procedure for dual mixed discretisations of elliptic boundary value problems. We complement our deliberation in chapter 4 with a closer look at the technical requirements, a finite element software should meet to fully exploit the potential of generalised hypercycle estimates. The use of multilevel solvers for constrained variational problems in a dual mixed formulation is discussed in section 4.3. In this context, we provide a proof of convergence for a new iteration scheme, that we may be employed either as a smoother or as a solver, for it admits a preconditioner. The description of numerical experiments in chapter 5 concludes this text.

Chapter 1

A general Framework for A posteriori Error Estimators based on Duality Techniques

Throughout the last thirty years, roughly speaking, a substantial number of numerical schemes has been developed which aim at supplying estimates for the error we necessarily incur when we employ computing machinery to find approximations to the solution of partial differential equations. Any attempt at assessing their respective amenities or disadvantages we deem futile. Hence, we have not arranged the following paragraphs with the intention of pitching a number of widely known and well approved numerical techniques against some new procedures of our own invention in order to show that our approach will yield more precise a posteriori error estimates than the established methods do. Let us instead inquire into the minimal requirements we have to impose on a variational problem such that we are still able to recover reliable information about the approximation error from the data of the problem and from its numerical solution.

We shall try and reach our goal in a somewhat implicit manner by attempting to exploit only the generic properties of the variational problem, when we derive our a posteriori estimates. Consequently, our treatment of the subject matter will be rather abstract: examples shall be supplied in the succeeding chapter. We want to stress, we are not going to sacrifice the usefulness of our results for the sake of the most general treatment possible. In fact our scope will be limited to the minimisation of such convex functionals, as they have already been studied by Fenchel [67, 68] in the middle of the last century - namely to problems of the form:

$$F(\Lambda x) \; + \; G(x) \quad \underset{x \in X}{\longrightarrow} \quad \inf$$

with F and G denoting convex functionals and Λ a continuous, linear operator. We have already endeavoured to justify in our introduction, why the uniform convexity of the functional F is a requirement we cannot dispense with, if we want to extract any information on the approximation error from the numerical solution. Therefore, we will not enter into these deliberations again. Our setting is sufficiently comprehensive to cover a substantial number of relevant problems.

The tool we will mainly employ in analysing a posteriori estimates for convex variational problems is a calculus commonly known as *Fenchel transform*. The first section of this chapter contains a compilation of assorted facts, chiefly about convex functions and the properties of the Fenchel transform. It is intended as a short primer and for referencing purposes. In the following section we present a detailed development of those generalised hypercycle estimates, we have briefly mentioned in the introduction. The last section offers an outlook how the mathematical technology discussed in section 1.2 can be extended to comprise computable a posteriori estimates for the output of linear functionals, which are applied to the numerical solution of the variational problem in order to gauge particular features of that approximation.

1.1 Preliminaries

In order to develop our theoretical framework for generalised hypercycle estimates in a possibly self-contained manner we will recapitulate a number of basic definitions and results from the field of convex analysis. As it is impossible to give a complete survey of the mathematics involved in the development of such estimates, a familiarity with the elementary concepts of topology and functional analysis is hereby assumed on the part of the reader.

1.1.1 Convex sets and paired spaces

We recall that a subset $C \subset X$ of some real linear space X is called *convex*, if for any two elements $x_1 \in C$ and $x_2 \in C$ and for any real parameter $\lambda \in [0, 1]$ the so called *convex combination*

$x_\lambda := \lambda x_1 + (1-\lambda)x_2$ is again contained in C. For any subset $C \subset X$ the *convex hull* abbreviated by $co\, C$ is defined as the smallest convex subset of X still containing C.

A *pairing* between two real linear spaces X and Y is a bilinear form $\langle \cdot, \cdot \rangle : X \times Y \longrightarrow \mathbb{R}$. Any locally convex topology on X is termed *compatible* with this pairing, if the functional

$$\langle \cdot, y \rangle \; : \; X \; \longrightarrow \; \mathbb{R}$$

is continuous for any $y \in Y$ and if every continuous linear functional on X can be represented in this way for some $y \in Y$. A compatible topology on Y can be defined mutatis mutandis. The spaces X and Y are said to be *paired*, once a particular pairing has been singled out and both X and Y have been equipped with compatible topologies with respect to that pairing.

Let X and X' denote two paired spaces. For any $x' \in X' \setminus \{0\}$ and any $\alpha \in \mathbb{R}$ the *half-space*

$$H_{x',\alpha} \; := \; \left\{ \, x \in X \, \middle| \, \langle x, x' \rangle \, \leq \, \alpha \, \right\}$$

may be introduced. We recall that any closed, convex subset of X can be represented as an intersection of such half-spaces. To be more specific let $C \subset X$ denote an arbitrary set and define $C' := \{ \, (x', \alpha) \in X' \times \mathbb{R} \, | \, C \subset H_{x',\alpha} \, \}$. The following identity is known [60] to hold:

$$cl(\, co\, C \,) \quad = \quad \bigcap_{(x',\alpha) \in C'} H_{x',\alpha} \; .$$

1.1.2 Convex and uniformly convex functions

A function $f : X \longrightarrow [-\infty, +\infty]$ shall be called *convex*, if its *epigraph* $epi\, f$ which is defined by

$$epi\, f \; := \; \left\{ \, (x, \sigma) \in X \times (-\infty, +\infty) \, \middle| \, f(x) \, \leq \, \sigma \, \right\}$$

is a convex subset of $X \times \mathbb{R}$ in the sense specified above. If the function f is merely defined on some subset $C \subset X$, the above definition will be applied to the *extension* $\tilde{f} : X \longrightarrow [-\infty, \infty]$ of the original function f, which is specified by:

$$\tilde{f}(x) \; := \; \begin{cases} f(x) & ; \quad x \in C \\ +\infty & ; \quad \text{else} \end{cases} \; .$$

In the following, we will tacitly identify the function f with its extension $\tilde{f}$. We note, that

$$dom\, f \; := \; \left\{ \, x \in X \, \middle| \, f(x) \, < \, +\infty \, \right\}$$

referred to as the *effective domain* of the function f is a convex subset of X, if f is convex. We shall term a function f *proper*, if its effective domain is not empty and $f(x) > -\infty$ holds for all $x \in X$. We remark that a proper function $f : X \longrightarrow (-\infty, +\infty]$ is convex if and only if $dom\, f$ is a convex set and f is a convex function relative to $dom\, f$ in the classical sense:

$$f(\, \lambda \, x_1 \, + \, (1-\lambda) \, x_2 \,) \quad \leq \quad \lambda \, f(x_1) \, + \, (1-\lambda) \, f(x_2) \quad ; \quad \lambda \in (0,1) \tag{1.1}$$

with arbitrary $x_1, x_2 \in dom\, f$. Should the left hand side of (1.1) be strictly smaller than the right hand side, the function f is called *strictly convex* relative to $dom\, f$.

The *convex hull* of some function $f : X \longrightarrow (-\infty, +\infty]$ will be denoted by $co\, f$. It is defined as the largest convex function, which is less or equal f in a pointwise sense. Geometrically, the epigraph of $co\, f$ can be obtained from the convex hull of $epi\, f$:

$$epi(co\, f) \quad = \quad \left\{ \, (x, \sigma) \in X \times \mathbb{R} \, \middle| \, (x, \rho) \in co\, (epi\, f) \; ; \; \rho > \sigma \, \right\} \; .$$

A proper, convex function $f : X \longrightarrow (-\infty, +\infty]$ shall be called *uniformly convex* at some point $y \in dom\, f$ if there is a nondecreasing function $\delta : \mathbb{R}_0^+ \longrightarrow \mathbb{R}_0^+$ with the following properties:

$$f\Big(\frac{x+y}{2} \Big) \quad \leq \quad \frac{1}{2} \, f(x) \; + \; \frac{1}{2} \, f(y) \; - \; \delta(\, \| x - y \| \,) \qquad ; \qquad x \in dom\, f$$

and $\delta(t) > 0$ for any argument $t > 0$. If the above inequality holds for any $y \in dom\, f$ and if δ can be specified irrespectively of the point y, the function f is termed *uniformly convex*. The *indicator function* $\chi_U : X \longrightarrow \{0, +\infty\}$ of some subset $U \subset X$ is defined by:

$$\chi_U(x) \;:=\; \begin{cases} 0 & ; \quad x \in U \\ +\infty & ; \quad \text{else} \end{cases} \quad .$$

We say, that f is uniformly convex on $U \subset X$, if the function $f + \chi_U$ is uniformly convex.

1.1.3 Lower-semicontinuous functions

A function $f : X \longrightarrow [-\infty, +\infty]$ is called *lower-semicontinuous* if its epigraph $epi\, f \subset X \times \mathbb{R}$ is closed. This concept allows for two constructions by which new functions can be derived from f. The *lower-semicontinuous hull* of f designated by $lsc\, f$ may be introduced as the largest lower-semicontinuous function, which is smaller or equal f in a pointwise sense. Hence, we note: $epi\,(lsc\, f) = cl(epi\, f)$. Equivalently, we may define:

$$(lsc\, f)(x) \;:=\; \liminf_{y \to x} f(y) \quad ; \quad x \in X \quad .$$

The *closure* of some function f, which will be denoted by $cl\, f$, differs but slightly from the lower-semicontinuous hull of f. It is defined as:

$$cl\, f \;:=\; \begin{cases} lsc\, f & ; \quad f > -\infty \\ -\infty & ; \quad \text{else} \end{cases} \quad .$$

A function f shall be called *closed*, if $f = cl\, f$ holds. Assume that f is convex. In this case also $lsc\, f$ and $cl\, f$ are convex functions. Furthermore, if $lsc\, f$ has a finite value at some point, then f and $lsc\, f$ are proper and $cl\, f = lsc\, f$ holds. Otherwise, $lsc\, f$ is of the form:

$$lsc\, f \;:=\; \begin{cases} -\infty & ; \quad x \in cl(dom\, f) \\ +\infty & ; \quad \text{else} \end{cases} \quad .$$

1.1.4 The Fenchel transform

Let X and Y denote two paired spaces which are related by the bilinear form $\langle \cdot, \cdot \rangle : X \times Y \longrightarrow \mathbb{R}$. With any function $f : X \longrightarrow [-\infty, +\infty]$ a *conjugate* function $f^* : Y \longrightarrow [-\infty, +\infty]$ defined as

$$f^*(y) \;:=\; \sup_{x \in X} \left\{ \langle x, y \rangle \,-\, f(x) \right\} \quad ; \quad y \in Y \tag{1.2}$$

may be associated. The function f^* is convex and closed by construction. In the same pattern a conjugate function $g^* : X \longrightarrow [-\infty, \infty]$ can be specified for any function $g : Y \longrightarrow [-\infty, \infty]$:

$$g^*(x) \;:=\; \sup_{y \in Y} \left\{ \langle x, y \rangle \,-\, g(y) \right\} \quad ; \quad x \in X \quad .$$

The mapping $f \longrightarrow f^*$ is called the *Fenchel transform*. We note, that $f^{**} = cl(co\, f)$ holds for any function f on X respectively Y. Hence, the Fenchel transform induces a one-to-one mapping between the closed convex functions on X and the closed convex functions on Y. The proof is simple and can be found e. g. in [128].

A function f is called *concave* if the extension of $-f$ is convex. Using this approach most of the concepts, that have been reviewed so far, can be translated easily to the case of concave functions. The definition introduced in the convex case is thereby applied to the function $-f$, then the result is reconverted by multiplying with -1. Thus *upper-semicontinuous* functions, *upper-semicontinuous hulls* and *(upper) closures* of concave functions can be specified. The exception to this rule is the Fenchel transform: Instead of considering $-(-f)^*$ as the conjugate of some concave function f we introduce a Fenchel transform $f \longrightarrow f_*$ in the concave sense:

$$f_*(y) \;:=\; \inf_{x \in X} \left\{ \langle x, y \rangle \,-\, f(x) \right\} \quad ; \quad y \in Y \quad .$$

Analogous results hold for convex and concave functions. Especially, we find $f_{**} = cl\, f$ for any concave function f on X respectively Y. Therefore, the operation $f \longrightarrow f_*$ induces a one-to-one mapping between the closed concave functions on X and the closed concave functions on Y. The conjugates of some function f on X in the convex and in the concave sense are related by:

$$-\, f_*(y) \;=\; (-f)^*(-y) \qquad ; \qquad y \in Y \;\; .$$

1.1.5 Subdifferentials

Let X and Y denote two linear spaces, that have been paired by means of the bilinear form $\langle \cdot, \cdot \rangle : X \times Y \longrightarrow \mathbb{R}$. The *subdifferential* $\partial f(x) \subset Y$ of an arbitrary function $f : X \longrightarrow [-\infty, +\infty]$ at some point $x \in X$ is defined as the set

$$\partial f(x) \;:=\; \left\{\, y \in Y \;\middle|\; f(x) + f^*(y) \,=\, \langle x, y \rangle \,\right\} \;\; .$$

With a view to the definition (1.2) of the Fenchel transform f^* we may state, that $y \in \partial f(x)$ holds if and only if $f(x)$ has a finite value and additionally:

$$f(x') \;\geq\; f(x) \,+\, \langle\, y, x' - x \,\rangle \qquad ; \qquad x' \in X \;\; .$$

Hence, we can identify each *subgradient* $y \in \partial f(x)$ with a continuous affine mapping, that is nowhere greater than the function f but coincides with it at the very point $x \in dom\, f$. We recall (see e. g. [60]) that a proper function f, which is convex and continuous over the interior of its effective domain, has a nonempty subdifferential $\partial f(x) \neq \emptyset$ for any $x \in dom\, f$. In case this function f is also *Gâteaux-differentiable* at the point $x_0 \in X$, that is:

$$y \in Y \qquad : \qquad \lim_{\mu \downarrow 0} \frac{f(x_0 + \mu x) \,-\, f(x_0)}{\mu} \;=\; \langle y, x \rangle \qquad ; \qquad x \in X \;\; .$$

Its subdifferential is known to consist of only one element: $\partial f(x_0) = \{y\}$. Conversely, if the convex function is continuous at the point $x \in dom\, f$ and there is only one subgradient $y \in \partial f(x)$ then f is Gâteaux-differentiable with $df(x) = y$.

1.1.6 Properties of uniformly convex functions

A uniformly convex function can be characterised in different ways, that are equivalent to the definition presented in section 1.2.2. One such definition reads: A proper, lower-semicontinuous and convex function $f : X \longmapsto (-\infty, +\infty]$ is uniformly convex at $x \in dom\, f$, if there exists a nondecreasing function $\delta : \mathbb{R}_0^+ \longrightarrow [0, +\infty]$ with the property $\delta(t) > 0$ for all $t > 0$, such that

$$f(\lambda x + (1-\lambda) y) \;\leq\; \lambda\, f(x) \,+\, (1-\lambda)\, f(y) \,-\, \lambda(1-\lambda)\, \delta(\|y - x\|) \qquad ; \qquad y \in dom\, f$$

holds for any value of $\lambda \in (0, 1)$. Another possible characterisation of the uniform convexity of f at the point $x \in dom\, f$ reads: For any $\epsilon > 0$ there is some $\delta > 0$, such that

$$f\Big(\frac{y+x}{2}\Big) \;\leq\; \frac{1}{2}\, f(y) \;+\; \frac{1}{2}\, f(x) \;-\; \delta \tag{1.3}$$

holds for any $y \in dom\, f$ with the property: $\|x - y\| \geq \epsilon$. Evidently, this latter definition is equivalent to the one presented in section 1.2.2. The former definition can be obtained from the latter in the following manner: We fix $\epsilon > 0$ and $\lambda \in (0, 0.5)$. If $y \in dom\, f$ satisfies $\|x - y\| \geq \epsilon$, the convexity of f and the estimate (1.3) warrant:

$$\begin{aligned}
f(\lambda x + (1-\lambda) y) \;&=\; f\Big(2\lambda\Big(\frac{x+y}{2}\Big) \,+\, (1-2\lambda) y\Big) \\
&\leq\; 2\,\lambda\, f\Big(\frac{x+y}{2}\Big) \,+\, (1-2\lambda)\, f(y) \\
&\leq\; \lambda\, f(x) \,+\, \lambda\, f(y) \,-\, 2\,\lambda\,\delta \,+\, (1-2\lambda) f(y) \\
&<\; \lambda\, f(x) \,+\, (1-\lambda)\, f(y) \,-\, 2\,\lambda\,(1-\lambda)\,\delta \;\; .
\end{aligned}$$

The very same estimate can be obtained for $\lambda \in (0.5, 1)$ by reversing the roles of x and y and replacing λ with $1 - \lambda$. We assume that X is a reflexive Banach space and $f : X \longrightarrow (-\infty, +\infty]$ a proper, lower semi-continuous function, such that the interior of $dom\, f$ is not empty. We will say, that a function $\delta : \mathbb{R}_0^+ \longrightarrow [0, +\infty]$ belongs to the set $\mathfrak{D}$, if it has the following properties: δ is lower-semicontinuous and convex, the interior of $dom\, \delta$ is not empty and $\delta(t) = 0$ implies $t = 0$. We define: $\partial f := \{\, (x, x^*) \in X \times X^* \,|\, x^* \in \partial f(x) \,\}$. If f is uniformly convex at $x \in dom\, f$, such that $\partial f(x) \neq \emptyset$ holds, and $x^* \in \partial f(x)$ is an arbitrary element, we may assert:

i) $\delta \in \mathfrak{D} \quad : \quad f(y) \;\geq\; f(x) + \langle y - x, x^* \rangle + \delta(\|y - x\|) \quad ; \quad y \in X$

ii) $\delta \in \mathfrak{D} \quad : \quad f^*(y^*) \;\leq\; f^*(x^*) + \langle x, y^* - x^* \rangle + \delta^*(\|y^* - x^*\|) \quad ; \quad y^* \in X^*$

iii) $\delta \in \mathfrak{D} \quad : \quad \delta(\|x - y\|) \;\leq\; \langle y - x, y^* - x^* \rangle \quad ; \quad (y, y^*) \in \partial f$

iv) f^* is Fréchet differentiable at $x^* \in \partial f(x)$.

v) There is a nondecreasing function $\phi : \mathbb{R}_0^+ \longrightarrow [0, +\infty]$, such that:

$$\lim_{t \downarrow 0} \phi(t) \;=\; 0 \qquad \wedge \qquad \|y - x\| \;\leq\; \phi(\|y^* - x^*\|) \quad ; \quad (y, y^*) \in \partial f \;\; .$$

A proof of the above statements can be found in [141]. Stronger results can been obtained, if the function f is uniformly convex. We will say, that a function $\delta \in \mathfrak{D}$ is contained in the subspace $\mathfrak{D}_0$, if the quotient $\delta(t)/t^2$ is bounded away from 0 in the limit $t \to \infty$. We contend:

vi) $\delta_0 \in \mathfrak{D}_0 \;:\; f(y) \;\geq\; f(x) + \langle y - x, x^* \rangle + \delta_0(\|y - x\|) \quad ; \quad (x, x^*) \in \partial f,\; y \in dom\, f$

vii) $\delta_0 \in \mathfrak{D}_0 \;:\; f^*(y^*) \leq f^*(x^*) + \langle x, y^* - x^* \rangle + \delta_0^*(\|y^* - x^*\|) \;;\; (x, x^*) \in \partial f,\; y^* \in dom\, f^*$

viii) $\delta_0 \in \mathfrak{D}_0 \;:\; \delta_0(\|x - y\|) \;\leq\; \langle y - x, y^* - x^* \rangle \quad ; \quad (x, x^*)\,,\, (y, y^*) \;\in\; \partial f$

ix) f^* is Fréchet equidifferentiable in the interior of $dom\, f$.

x) There is a nondecreasing function $\phi_0 : \mathbb{R}_0^+ \longrightarrow [0, +\infty]$, such that:

$$\lim_{t \downarrow 0} \phi_0(t) \;=\; 0 \qquad \wedge \qquad \|y - x\| \;\leq\; \phi_0(\|y^* - x^*\|) \quad ; \quad (x, x^*)\,,\, (y, y^*) \;\in\; \partial f \;\; .$$

Again, for all of these assertions proofs are supplied in [141]. As a corollary of the assertion vi) we may claim, that every uniformly convex functional $f : X \longrightarrow (-\infty, +\infty]$ is coercive.

1.2 Error Estimates in the Energy Norm

The following paragraphs constitute the core of chapter 1 and summarise the major part of our research efforts, if we take only the theoretical results into account. Under the sole assumption, that the numerical solution of our variational problem is an admissible function, we derive reliable bounds on the approximation error which is measured in the so called energy norm. In the first section we specify the variational setting we are going to consider and define our notation. In the following section we introduce a dual formulation with the help of the Fenchel transform. Combining the primal with the dual statement we thus obtain an abstract error bound, which serves as a starting point for all further manipulations. Since the abstract error estimate relies on the dual statement of the variational problem an additional admissibility constraint has been introduced. This dual admissibility constraint is taken into account by a suitable penalisation, which we derive in section 1.2.3 from certain generic properties of uniformly convex functionals, respectively from those of their Fenchel conjugates. The emerging error bounds may be viewed as consisting of two residual expressions, which correspond to the necessary optimality conditions of the primal formulation, and a mixed term, which contains information on both residuals. In section 1.2.4 we show, how these contributions can be decoupled. In the following section an alternative approach is discussed, which ultimately leads to a sharper if more involved error bound. In section 1.2.6 a number of properties are compiled, the generalised hypercycle estimates feature. These results are summarised in the final section 1.2.7.

1.2.1 Statement of the variational formulation

Let X and Y denote reflexive Banach spaces and $\Lambda : X \longrightarrow Y$ a continuous linear operator. The spaces X^* and Y^* are paired with X respectively Y. The corresponding pairings are denominated $\langle \cdot, \cdot \rangle_X : X \times X^* \longrightarrow \mathbb{R}$ and $\langle \cdot, \cdot \rangle_Y : Y \times Y^* \longrightarrow \mathbb{R}$. We assume, that Λ has a continuous transposed $\Lambda^* : Y^* \longrightarrow X^*$, which is characterised by:

$$\left\langle x \,,\, \Lambda^* y^* \right\rangle_X \;:=\; \left\langle \Lambda x \,,\, y^* \right\rangle_Y \qquad ; \qquad y^* \in Y^* \;\; .$$

Let $F : Y \longrightarrow \mathbb{R}$ and $G : X \longrightarrow (-\infty, +\infty]$ denote two convex functionals. Let us assume, that the first functional is uniformly convex, while the latter is merely proper and continuous over its effective domain. We require, that the operator Λ be coercive:

$$\lambda_0 > 0 \qquad : \qquad \| x \|_\Lambda \;:=\; \| \Lambda x \|_Y \;\geq\; \lambda_0 \, \| x \|_X \qquad ; \qquad x \in X \;\; . \tag{1.4}$$

Hence, another norm is defined on the space X by the left hand side of the above inequality, which below we will refer to as the *energy norm*. We will consider the minimisation problem

$$J(x, \Lambda x) \;:=\; F(\Lambda x) \;+\; G(x) \quad \underset{x \in X}{\longrightarrow} \quad \inf \;\; . \tag{1.5}$$

The existence of an unique solution $x_0 \in X$ to this primal formulation is a well known fact (see e. g. [60]), for the functional F and therewith also J is coercive. With a view to section 1.1.6 we can state furthermore, that there is a forcing function $\phi \in \mathfrak{D}_0$, such that

$$2 \, F\big(\tfrac{y_1 + y_2}{2}\big) \;\leq\; F(y_1) \;+\; F(y_2) \;-\; \phi(\, \| y_1 - y_2 \|_Y) \tag{1.6}$$

holds for any two elements $y_1, y_2 \in Y$.

1.2.2 An abstract a posteriori error estimate

Our assumptions on the functional J ensure there is an unique minimiser $x_0 \in X$. Let us suppose that we have obtained an approximation $x \in X$ to this element by some numerical procedure. We may bound the approximation error by:

$$2 \;\phi(\, \| x - x_0 \|_\Lambda \,) \;\leq\; J(x, \Lambda x) \;-\; J(x_0, \Lambda x_0) \;\; .$$

To substantiate our claim let us recapitulate the results presented in section 1.1.6. We select an arbitrary parameter $\eta \in (0, 1)$ and introduce the abbreviation $y := \eta \, x_0 + (1 - \eta) \, x$. Thanks to (1.6) and the convexity of G the following inequality holds:

$$\begin{aligned} 2 \, \eta \, (1 - \eta) \; \phi(\, \| x - x_0 \|_\Lambda \,) \;&\leq\; \eta \, F(\Lambda x_0) \;+\; (1 - \eta) \, F(\Lambda x) \;-\; F(\Lambda y) \\ &\leq\; \eta \, J(x_0, \Lambda x_0) \;+\; (1 - \eta) \, J(x, \Lambda x) \;-\; J(y, \Lambda y) \\ &\leq\; (1 - \eta) \; (\; J(x, \Lambda x) \;-\; J(x_0, \Lambda x_0) \,) \;\; . \end{aligned}$$

Dividing the above equation by $(1 - \eta)$ and proceeding to the limit $\eta \to 1$ finishes the proof. As both F and G are convex and closed, we can represent the functional $J : X \times Y \longrightarrow \mathbb{R}$ with the help of its conjugate $J^* : X^* \times Y^* \longrightarrow \mathbb{R}$. Thereby, the pairing for the product spaces $X \times Y$ and $X^* \times Y^*$ may simply be defined as a sum: $\langle x, x^* \rangle_X + \langle y, y^* \rangle_Y$. Accordingly, we find:

$$\begin{aligned} \inf_{x \in X} J(x, \Lambda x) \;&=\; \inf_{x \in X} \Big\{ \sup_{x^* \in X^*} \sup_{y^* \in Y^*} \big\{ \, \langle x, x^* \rangle \;+\; \langle \Lambda x, y^* \rangle \;-\; J^*(x^*, y^*) \, \big\} \Big\} \\ &\geq\; \sup_{x^* \in X^*} \sup_{y^* \in Y^*} \Big\{ \inf_{x \in X} \big\{ \, \langle x, x^* + \Lambda^* y^* \, \rangle \;-\; J^*(x^*, y^*) \, \big\} \Big\} \\ &=\; \sup_{y^* \in Y^*} \big\{ \, -J^*(-\Lambda^* y^*, y^*) \, \big\} \;\geq\; -J^*(-\Lambda^* z^*, z^*) \end{aligned} \tag{1.7}$$

for any element $z^* \in Y^*$. The conjugate functional J^* can be decomposed in the following way:

$$J^*(x^*, y^*) \;=\; \sup_{x \in X} \sup_{y \in Y} \big\{ \, \langle x, x^* \rangle + \langle y, y^* \rangle \;-\; F(y) - G(x) \, \big\} \;=\; F^*(y^*) \;+\; G^*(x^*)$$

with $x^* \in X^*$ and $y^* \in Y^*$. By combining these results an abstract bound on the approximation error is obtained, that consists of two distinct contributions, both of them being positive:

$$2\,\phi(\,\|x - x_0\|_\Lambda\,) \;\leq\; M_F(\Lambda x, y^*) \;+\; M_G(x, -\Lambda^* y^*) \qquad ; \qquad y^* \in Y^* \;. \tag{1.8}$$

Said contributions feature the same mathematical structure. Actually, they read:

$$\begin{aligned} M_F(y, y^*) \;&:=\; F(y) \;+\; F^*(y^*) \;-\; \langle y, y^*\rangle_Y \qquad ; \qquad y \in Y,\; y^* \in Y^* \;, \\ M_G(x, x^*) \;&:=\; G(x) \;+\; G^*(x^*) \;-\; \langle x, x^*\rangle_X \qquad ; \qquad x \in X,\; x^* \in X^* \;. \end{aligned}$$

1.2.3 Towards a computable error majorant

While the first part of the abstract error estimate (1.8) is always finite, the second part of the estimate need not necessarily be so. Even if the requirement $x \in dom\,G$ is met, $M_G(x, \Lambda^* y^*)$ can still be infinite depending on our choice of the parameter $y^* \in Y^*$. Hence, we may be reduced to considering a constrained minimisation problem of the form

$$M_F(\Lambda x, -z^*) \;+\; M_G(x, \Lambda^* z^*) \quad \underset{z^* \in Z^*}{\longrightarrow} \quad \inf \tag{1.9}$$

with the convex set Z being defined by: $Z^* := (\Lambda^*)^{-1}(dom\,G^*)$ merely to obtain a meaningful bound on the approximation error. As problem (1.9) will be more difficult to treat in most applications than the primary minimisation task, we conclude, that the formula (1.8) needs to be revised in order to become a widely applicable error estimator. With a view to paragraph 1.1.6 we may state, that the conjugate functional F^* is Fréchet-differentiable. Furthermore, we may determine a convex function $h_0 \in \mathfrak{D}_0$, such that:

$$\begin{aligned} M_F(\Lambda x, -z^*) \;&=\; F(\Lambda x) \;+\; F^*(-z^*) \;-\; \langle \Lambda x, z^*\rangle_Y \\ &=\; F(\Lambda x) \;+\; F^*(y^*) \;-\; \langle \Lambda x, y^*\rangle_Y \;+\; \langle \Lambda x, y^* + z^*\rangle_Y \;+\; F^*(-z^*) - F^*(y^*) \\ &=\; M_F(\Lambda x, y^*) \;+\; \langle\, \Lambda x - dF^*(y^*)\,,\, y^* + z^*\,\rangle_Y \;+ \\ &\qquad +\quad F^*(-z^*) \;-\; F^*(y^*) \;-\; \langle\, dF^*(y^*)\,,\, -z^* - y^*\,\rangle_Y \qquad (1.10) \\ &\leq\; M_F(\Lambda x, y^*) \;+\; \langle\, \Lambda x - dF^*(y^*)\,,\, y^* + z^*\,\rangle_Y \;+\; h_0^*(\,\|\,y^* + z^*\|_{Y^*}) \end{aligned}$$

holds for any two elements $y^*, z^* \in Y^*$. A simple calculation shows, that h_0^* is nonnegative and that $h_0^*(0) = 0$ holds. Let us define the auxiliary function $h : Y \longrightarrow \mathbb{R}_0^+$:

$$h(y) \;:=\; \sup_{t \geq 0} \left\{\, t\,\|y\|_Y \;-\; h_0^*(t) \,\right\} \qquad ; \qquad y \in Y \;. \tag{1.11}$$

Since this function is the supremum of a family of convex and closed functions, it is convex and closed itself. We note: $h^*(y^*) = h_0^*(\|y^*\|_{Y^*})$ for any element $y^* \in Y^*$.

Invoking (1.10) we may bound the first part of the error majorant in (1.9) using an arbitrary element $y^* \in Y^*$ instead of $-z \in Z^*$. Since we will eventually minimise the error majorant with respect both to z^* and y^* we have not given anything away as yet. The next step in our analysis is more momentous. With a view to the optimality conditions of problem (1.9), namely:

$$-\,z^* \;\in\; \partial F(\Lambda x) \qquad \wedge \qquad \Lambda^* z^* \;\in\; \partial G(x) \tag{1.12}$$

it seems most reasonable to consider those contributions to the error majorant, that may be conceived as residual terms with respect to the first of the above duality relations, separately from those contributions that correspond to the second one. While $M_F(\Lambda x, y^*)$ is associated with the first duality relation and $M_G(x, \Lambda^* z^*) + h^*(y^* + z^*)$ with the second one, the role of the expression $\langle \Lambda x - dF^*(y^*), y^* + z^*\rangle_Y$ is ambiguous. Employing an estimate of the form:

$$\langle\, \Lambda x - dF^*(y^*)\,,\, y^* + z^*\,\rangle_Y \;\leq\; \frac{1}{2}\,\|\,\Lambda x - dF^*(y^*)\,\|_Y^2 \;+\; \frac{1}{2}\,\|\,y^* + z^*\,\|_{Y^*}^2 \tag{1.13}$$

we may preserve the dichotomy present in the abstract error bound (1.8) but we loose some of its precision at the same time. If we aim at a sharp estimate, we will have to treat the left hand side of (1.13) along with $M_G(x, \Lambda^* z^*) + h^*(y^* + z^*)$. In that latter case the resulting error majorant will no longer consist of two distinct parts that can be easily matched against the duality relations (1.12). Consequently, it may be much harder to evaluate.

1.2.4 Separating primal and dual variables

The approach we will discuss below was proposed in [125] in dealing with the special case that G is a continuous linear functional. It is evident, however, that its main idea is also applicable in the more general case we have specified in section 1.2.1. Let us introduce a proper function $H : Y \longrightarrow \mathbb{R}$. Our choice of this function is arbitrary, but should depend on the actual properties of the functionals F and G. By definition H and its conjugate H^* satisfy the inequality:

$$\langle y, y^* \rangle_Y \;\leq\; H(y) \,+\, H^*(y^*) \quad ; \quad y \in Y \,,\; y^* \in Y^* \;. \tag{1.14}$$

Using this inequality to bound the second expression on the right hand side of (1.10) we arrive at a new error majorant by combining (1.8) with (1.10):

$$2\,\phi(\,\|x - x_0\|_\Lambda\,) \;\leq\; \hat{M}_F(\Lambda x, y^*) \,+\, \hat{M}_G(x, y^*) \quad ; \quad y^* \in Y^* \;. \tag{1.15}$$

The revised contributions $\hat{M}_F$ and $\hat{M}_G$ to this error estimate are defined by

$$\begin{aligned} \hat{M}_F(\Lambda x, y^*) \;&:=\; M_F(\Lambda x, y^*) \;+\; H(\,\Lambda x - dF^*(y^*)\,) \;\;, \\ \hat{M}_G(x, y^*) \;&:=\; \inf_{z^* \in Y^*} \left\{\, M_G(x, \Lambda^* z^*) \,+\, H^*(y^* + z^*) \,+\, h^*(y^* + z^*) \,\right\} \end{aligned}$$

for any parameter $y^* \in Y^*$. In order to evaluate $\hat{M}_G$ let us introduce the following Lagrangian:

$$L(z^*; y, z, s) \;:=\; G(x) - G(s) - \langle \Lambda(x - s), z^* \rangle_Y + \langle y + z, y^* + z^* \rangle_Y - H^{**}(z) - h(y)$$

with $z^* \in Y^*$ and $y, z \in Y$ respectively $s \in X$. Obviously, this function is convex and continuous in its first argument, as it is actually an affine mapping. Accordingly, we find:

$$y, z \in Y \,,\; s \in X \quad : \quad \lim_{\|z^*\|_{Y^*} \to \infty} L(z^*; y, z, s) \;=\; \infty \quad .$$

With respect to $y, z \in Y$ and $s \in X$ the Lagrangian can be identified as a sum of a continuous linear mapping and a concave, upper-semicontinuous function. As such $L(z^*; \cdot, \cdot, \cdot)$ is concave and upper-semicontinuous itself for any $z^* \in Y^*$. Though for theses reasons L need not necessarily have a saddle-point (see e. g. [60] proposition 2.3 in chapter VI), we still may state:

$$\inf_{z^* \in Y^*} \left\{\, \sup_{y,z \in Y} \sup_{s \in X} L(z^*; y, z, s) \,\right\} \;=\; \sup_{y,z \in Y} \sup_{s \in X} \left\{\, \inf_{z^* \in Y^*} L(z^*; y, z, s) \,\right\} \;. \tag{1.16}$$

By construction of L we can recover the left hand side of (1.16) in the following manner:

$$\begin{aligned} \hat{M}_G(x, y^*) \;&=\; \inf_{z^* \in Y^*} \Big\{\, G(x) + G^*(\Lambda^* z^*) - \langle \Lambda x, z^* \rangle_Y + H^*(y^* + z^*) + h^*(y^* + z^*) \,\Big\} \\ &=\; \inf_{z^* \in Y^*} \Big\{\, G(x) \,+\, \sup_{s \in X} \big\{\, \langle s, \Lambda^* z^* \rangle_X - G(s) \,\big\} \,-\, \langle \Lambda x, z^* \rangle_Y \quad + \\ &\qquad\qquad + \quad \sup_{y,z \in Y} \big\{\, \langle\, y + z, y^* + z^* \rangle_Y \,-\, H^{**}(z) - h(y) \,\big\} \,\Big\} \\ &=\; \inf_{z^* \in Y^*} \left\{\, \sup_{y,z \in Y} \sup_{s \in X} L(z^*; y, z, s) \,\right\} \;. \end{aligned} \tag{1.17}$$

Since the Lagrangian L is affine in its first argument, its infimum with respect to z^* will only be finite, if the dependency on z^* can be dropped altogether. Hence, we may define the set:

$$Y_{y,s} \;:=\; \left\{\, z \in Y \;\middle|\; \Lambda(x - s) \,=\, y + z \,\right\} \quad ; \quad y \in Y,\, s \in X \;.$$

Using this definition the infimum of L with respect to $z^* \in Y^*$ can be written as:

$$\inf_{z^* \in Y^*} L(z^*; y, z, s) \;=\; \begin{cases} G(x) - G(s) + \langle y + z, y^* \rangle_Y - H^{**}(z) - h(y) & ; \quad z \in Y_{y,s} \\ -\infty & ; \quad \text{else} \;. \end{cases}$$

To simplify the notation we introduce another function $\hat{L} : Y \times X \longrightarrow \mathbb{R}$:

$$\begin{aligned} \hat{L}(y,s) \;&:=\; \sup_{z \in Y} \Big\{ \inf_{z^* \in Y^*} L(z^*; y, z, s) \Big\} \;=\; \sup_{z \in Y_{y,s}} \Big\{ \inf_{z^* \in Y^*} L(z^*; y, z, s) \Big\} \\ &=\; G(x) - G(s) + \langle \Lambda(x-s), y^* \rangle_Y - H^{**}(\Lambda(x-s) - y) - h(y) \quad . \end{aligned}$$

If $x \in dom G$ holds, we find $\partial G(x) \neq \emptyset$ by construction of G. In this case we may bound $\hat{L}$ by:

$$\hat{L}(y,s) \;\leq\; \langle x - s, \Lambda^* y^* + x^* \rangle_X - H^{**}(\Lambda(x-s) - y) - h(y) \quad ; \quad x^* \in \partial G(x) \tag{1.18}$$

for arbitrary arguments $y \in Y$ and $s \in X$. We continue our analysis of $\hat{M}_G(x, y^*)$ and define:

$$K(s) \;:=\; \inf_{y \in Y} \big\{ \, H^{**}(\Lambda s - y) \,+\, h(y) \, \big\} \quad ; \quad s \in X \;. \tag{1.19}$$

For any subgradient $x^* \in \partial G(x)$ we can now estimate the supremum of $\hat{L}$ with respect to $y \in Y$:

$$\sup_{y \in Y} \hat{L}(y,s) \;\leq\; \langle x - s, \Lambda^* y^* + x^* \rangle_X \;-\; K(x-s) \quad ; \quad s \in X \;. \tag{1.20}$$

Combining (1.16) and (1.17) with the estimates (1.18) and (1.20) we conclude eventually:

$$\hat{M}_G(x, y^*) \;=\; \sup_{s \in X} \big\{ \sup_{y \in Y} \hat{L}(y,s) \big\} \;\leq\; \sup_{s \in X} \big\{ \, \langle x - s, \Lambda^* y^* + x^* \rangle_X - \, K(x-s) \, \big\} \;.$$

Therewith, the following bound for the approximation error $\|x - x_0\|_\Lambda$ has been derived:

$$2\,\phi(\,\|x - x_0\|_\Lambda\,) \;\leq\; \hat{M}_F(\Lambda x, y^*) \,+\, \inf_{x^* \in \partial G(x)} K^*(\Lambda^* y^* + x^*) \quad ; \quad y^* \in Y^* \;. \tag{1.21}$$

As far as that will be possible for as abstract an estimate as (1.21) we will discuss the properties of the above error majorant in a dedicated section below.

1.2.5 The second duality relation revisited

Starting from the error estimate (1.10) we will give an improved error bound compared to (1.21). We have already hinted at the necessity to consider primal and dual variables and their respective contributions to the error majorant simultaneously. Accordingly, we define a Lagrangian

$$L_y(z^*, z, s) \;:=\; G(x) - G(s) \,+\, \langle \Lambda s + z - y, z^* \rangle_Y \,+\, \langle \Lambda x + z - y, y^* \rangle_Y \,-\, h(z)$$

with $z^* \in Y^*$, $z \in Y$ and $s \in X$. The element $y \in Y$ is but an arbitrary parameter. Invoking the very same arguments we have employed in the previous section we can verify:

$$\inf_{z^* \in Y^*} \Big\{ \sup_{z \in Y} \sup_{s \in X} L_y(z^*; z, s) \Big\} \;=\; \sup_{z \in Y} \sup_{s \in X} \Big\{ \inf_{z^* \in Y^*} L_y(z^*; z, s) \Big\} \tag{1.22}$$

for any $y \in Y$. We proceed by identifying the right side of the above equation:

$$\begin{aligned} \hat{M}_G(y; x, y^*) \;&:=\; \inf_{z^* \in Y^*} \Big\{ M_G(x, \Lambda^* z^*) \,+\, \langle \Lambda x - y, y^* + z^* \rangle_Y \,+\, h^*(y^* + z^*) \Big\} \\ &=\; \inf_{z^* \in Y^*} \Big\{ G(x) + G^*(\Lambda^* z^*) + \langle \Lambda x - y, y^* \rangle_Y - \langle y, z^* \rangle_Y + h^*(y^* + z^*) \Big\} \\ &=\; \inf_{z^* \in Y^*} \Big\{ G(x) \,+\, \sup_{s \in X} \big\{ \, \langle s, \Lambda^* z^* \rangle_X \,-\, G(s) \, \big\} \,-\, \langle y, z^* \rangle_Y + \langle \Lambda x - y, y^* \rangle_Y \\ &+ \quad \sup_{z \in Y} \big\{ \, \langle \, z, y^* + z^* \rangle_Y \,-\, h(z) \, \big\} \Big\} \;=\; \inf_{z^* \in Y^*} \Big\{ \sup_{z \in Y} \sup_{s \in X} L_y(z^*; z, s) \Big\} \;. \end{aligned}$$

To evaluate the right hand side of (1.22) we compute the infimum of L_y with respect to $z^* \in Y^*$:

$$\inf_{z^* \in Y^*} L_y(z^*; z, s) \;=\; \begin{cases} G(x) - G(s) + \langle \Lambda x + z - y, y^* \rangle_Y - h(z) & ; \quad -z \in Y_{y, x+s} \\ -\infty & ; \quad \text{else} \;. \end{cases}$$

In analogy to the procedure we have applied previously we introduce a function $\hat{L}_y : X \longrightarrow \mathbb{R}$:

$$\hat{L}_y(s) \;:=\; \sup_{z \in Y} \Big\{ \inf_{z^* \in Y^*} L_y(z^*; z, s) \Big\} \;=\; G(x) - G(s) + \langle x - s, \Lambda^* y^* \rangle_Y - h(y - \Lambda s) \;.$$

Under the assumption $x \in dom\, G$ we can bound this auxiliary function by:

$$\hat{L}_y(s) \;\leq\; \langle x - s, \Lambda^* y^* + x^* \rangle_X \;-\; h(y - \Lambda s) \quad ; \quad x^* \in \partial G(x) \;. \tag{1.23}$$

Hereby, $s \in X$ and $y \in Y$ are arbitrary elements. We define for any $y \in Y$:

$$K_y(s) \;:=\; h(\, y \,+\, \Lambda s \,) \quad ; \quad s \in X \;.$$

Employing this new function K_y and combining the results (1.22) and (1.23) we find:

$$\begin{aligned} \hat{M}_G(y; x, y^*) \;=\; \sup_{s \in X} \hat{L}_y(s) \;&\leq\; \sup_{s \in X} \big\{ \langle x - s, \Lambda^* y^* + x^* \rangle_X \,-\, K_{y - \Lambda x}(x - s) \big\} \\ &=\; K^*_{y - \Lambda x}(\Lambda^* y^* + x^*) \quad ; \quad x^* \in \partial G(x) \;. \end{aligned}$$

An a posteriori estimate for the error $\|x - x_0\|_\Lambda$ can now be obtained by choosing $y = dF^*(y^*)$:

$$2\, \phi(\, \|x - x_0\|_\Lambda \,) \;\leq\; M_F(\Lambda x, y^*) \;+\; \inf_{x^* \in \partial G(x)} K^*_{dF^*(y^*) - \Lambda x}(\Lambda^* y^* + x^*) \tag{1.24}$$

with $y^* \in Y^*$ being arbitrary. Since both majorants (1.21) and (1.24) have been derived by the very same device, namely by bounding the difference $G(x) - G(s)$ in accordance with

$$G(x) \;-\; G(s) \;\leq\; \langle\, x - s \,,\, x^* \,\rangle_X \quad ; \quad x^* \in \partial G(x) \;,$$

the result (1.24) will provide a sharper estimate for the approximation error than the one we have obtained in the previous section, if we use the same parameter $y^* \in Y^*$ in both cases.

1.2.6 General features of the error majorants

Below we will discuss those features of the a posteriori error estimates (1.21) and (1.24) that can be inferred immediately from our assumptions on the minimisation problem (1.5). For our convenience let us introduce the following abbreviations:

$$\hat{M}_K(x, y^*) \;:=\; \hat{M}_F(\Lambda x, y^*) \;+\; \inf_{x^* \in \partial G(x)} K^*(\Lambda^* y^* + x^*) \;, \tag{1.25}$$

$$M_K(x, y^*) \;:=\; M_F(\Lambda x, y^*) \;+\; \inf_{x^* \in \partial G(x)} K^*_{dF^*(y^*) - \Lambda x}(\Lambda^* y^* + x^*) \tag{1.26}$$

with $x \in dom\, G$ and $y^* \in Y^*$. Since the definition (1.19) of the function K as well as that of K_y involves the operator $\Lambda : X \longrightarrow Y$, both of the above error majorants may still not be amenable to a numerical evaluation, though they are finite for any choice of $y^* \in Y^*$. In such a case, a suitable minorant $\tilde{K} : \tilde{X} \longrightarrow \mathbb{R}$ for K respectively K_y may be introduced. As X can be a true subset of $\tilde{X}$, it is necessary to impose some restrictions on the dual variable y^* in order to ensure $\{\Lambda^* y^*\} + \partial G(x) \subset \tilde{X}^*$. Fixing a suitable subset $\tilde{Y}^* \subset Y^*$ let us define:

$$\tilde{M}_K(x, y^*) \;:=\; \hat{M}_F(\Lambda x, y^*) \;+\; \inf_{x^* \in \partial G(x)} \tilde{K}^*(\Lambda^* y^* + x^*) \quad ; \quad y^* \in \tilde{Y}^* \;. \tag{1.27}$$

The minimal requirement we have to impose on any sensible error estimator is its ability to indicate, if we have found the true solution. Hence, we must verify:

$$\inf_{y^* \in Y^*} \big\{\, \hat{M}_K(x_0, y^*) \,\big\} \;=\; \inf_{y^* \in Y^*} \big\{\, M_K(x_0, y^*) \,\big\} \;=\; 0 \;. \tag{1.28}$$

In the case of the error majorants (1.25) and (1.26) it is comparatively easy to see that the requirement (1.28) is indeed met. Though the haven't made any use of the fact at the time, we introduced the first a posteriori error estimate in section 1.2.2, there is a maximisation problem related to the minimisation problem (1.5) in terms of the conjugate functional J^*:

$$-J^*(\Lambda^* y^*, -y^*) \;=\; -F^*(-y^*) \,-\, G^*(\Lambda^* y^*) \qquad \underset{y^* \in Y^*}{\longrightarrow} \qquad \sup \;\; .$$

Our assumption on the functionals F and G ensure, that this so called dual problem with respect to (1.5) features at least one solution $y_0^* \in Y^*$, such that the last estimate in (1.7) is in fact an equality (see e. g. §4 chapter III in [60]). Such a dual solution y_0^* is characterised by:

$$-\Lambda^* y_0^* \;\in\; \partial G(x_0) \qquad \wedge \qquad y_0^* \;\in\; \partial F(\Lambda x_0) \;\; .$$

Setting $y^* = y_0^*$ and exploiting the fact that $x^* := -\Lambda^* y_0^*$ is contained in $\partial G(x_0)$ we find:

$$\hat{M}_K(x_0, y_0^*) \;\leq\; M_F(\Lambda x_0, y_0^*) \,+\, H(\Lambda x_0 - dF^*(y_0^*)) \,+\, K^*(0) \;=\; H(0) \,+\, K^*(0) \;\; . \tag{1.29}$$

Hereby, we recall that $y_0^* \in \partial F(\Lambda x_0)$ implies $\Lambda x_0 \in \partial F^*(y_0^*)$ by definition of the subdifferential. For the same reasons we can bound the other error majorant M_K by:

$$M_K(x_0, y_0^*) \;\leq\; \inf_{x^* \in \partial G(x)} K_0^*(\Lambda^* y_0^* + x^*) \;\leq\; (h \circ \Lambda)^*(0) \;=\; h^*(0) \;\; . \tag{1.30}$$

Since $h^*(0) = 0$ holds by construction, this proves (1.28) in the case of the error majorant M_K. To find a bound for the right hand side of (1.29) we will derive a majorant for the function K^*. With a view to (1.4) we find for any $y^* \in Y^*$ and any $x^* \in X^*$:

$$\begin{aligned}
K^*(\Lambda^* y^* + x^*) \;&=\; \sup_{s \in X} \Big\{ \, \langle s, \Lambda^* y^* + x^* \rangle_X \,-\, \inf_{y \in Y} \big\{ \, H^{**}(\Lambda s - y) + h(y) \, \big\} \, \Big\} \\
&\leq\; \sup_{s \in X} \sup_{y \in Y} \Big\{ \, \langle \Lambda s, y^* \rangle_Y \,+\, \|s\|_X \|x^*\|_{X^*} \,-\, H^{**}(\Lambda s - y) \,-\, h(y) \, \Big\} \\
&\leq\; \sup_{s \in X} \sup_{y \in Y} \Big\{ \, \langle \Lambda s, y^* \rangle_Y \,+\, \lambda_0^{-1} \|\Lambda s\|_Y \|x^*\|_{X^*} \,-\, H^{**}(\Lambda s - y) \,-\, h(y) \, \Big\} \\
&\leq\; \sup_{z \in Y} \sup_{y \in Y} \Big\{ \, \langle z + y, y^* \rangle_Y \,+\, \lambda_0^{-1} \| \, z + y \, \|_Y \|x^*\|_{X^*} \,-\, H^{**}(z) \,-\, h(y) \, \Big\} \\
&\leq\; \sup_{y \in Y} \Big\{ \, \langle y, y^* \rangle_Y \,+\, \lambda_0^{-1} \| \, y \, \|_Y \|x^*\|_{X^*} \,-\, h(y) \, \Big\} \qquad (1.31) \\
&+\; \sup_{z \in Y} \Big\{ \, \langle z, y^* \rangle_Y \,+\, \lambda_0^{-1} \| \, z \, \|_Y \|x^*\|_{X^*} \,-\, H^{**}(z) \, \Big\} \;=:\; K_1^* \,+\, K_2^* \;\; .
\end{aligned}$$

Setting $x^* = 0$ and $y^* = 0$ we infer from in the above inequality and from (1.29):

$$\hat{M}_K(x_0, y_0^*) \;\leq\; H(0) \,+\, H^*(0) \;\; . \tag{1.32}$$

The result qualifies our choice of the function H to be used in the estimate (1.14). In order to meet (1.28) we must require the right hand side of (1.32) to vanish. For all practical purposes, however, the assumption $H(0) + H^*(0) = 0$ is not too restrictive. It is equivalent to:

$$H(0) \;=\; -H^*(0) \;=\; -\sup_{z \in Y} \{ \, -H(z) \, \} \;=\; \inf_{z \in Y} H(z) \;\; .$$

Let us assume in the following, that H and H^* are locally Lipschitz continuous functions. Let us suppose furthermore that the effective domain of H^* is the whole of Y^*. We conclude:

$$J(c) \;:=\; \sup_{\|y^*\| \leq c} H^*(y^*) \;=\; \sup_{y \in Y} \sup_{\|y^*\| \leq c} \big\{ \langle y, y^* \rangle_Y - H^{**}(y) \big\} \;=\; \sup_{y \in Y} \big\{ c \, \|y\|_Y - H^{**}(y) \big\} \;<\; +\infty$$

for any $c > 0$. In order to bound the first expression on the right hand side of (1.31) we recall that the mapping h can be expressed as a function of the norm of its argument:

$$\begin{aligned} K_1^* &= \sup_{y\in Y}\Big\{ \langle y, y^*\rangle_Y + \lambda_0^{-1}\,\|\,y\,\|_Y \|x^*\|_{X^*} - \sup_{t\geq 0}\{\, t\,\|y\|_Y - h_0^*(t)\,\}\Big\} \\ &\leq \inf_{t\geq 0}\sup_{r\geq 0}\Big\{ r \sup_{\|y\|=1}\langle y, y^*\rangle_Y + \lambda_0^{-1} r\,\|x^*\|_{X^*} - t\,r + h_0^*(t)\,\}\Big\} \\ &= \inf_{t\geq 0}\sup_{r\geq 0}\Big\{ r\,(\,\|y^*\|_{Y^*} + \lambda_0^{-1}\|x^*\|_{X^*} - t\,) + h_0^*(t)\Big\} \quad . \end{aligned}$$

The second expression on the right hand side of (1.31) may be controlled by:

$$K_2^* \leq \sup_{z\in Y}\Big\{\|z\|_Y(\,\|y^*\|_{Y^*} + \lambda_0^{-1}\|x^*\|_{X^*}) - H^{**}(z)\Big\} = J(\,\|y^*\|_{Y^*} + \lambda_0^{-1}\|x^*\|_{X^*}) \ .$$

Combining the above results we find the function K^* to be bounded by:

$$K^*(\Lambda^* y^* + x^*) \leq h_0^*(\,\|y^*\|_{Y^*} + \lambda_0^{-1}\,\|x^*\|_{X^*}\,) + J(\,\|y^*\|_{Y^*} + \lambda_0^{-1}\,\|x^*\|_{X^*}\,)$$

whereby $x^* \in X^*$ and $y^* \in Y^*$ are arbitrary. Furthermore, we note that K^* is a convex function by construction. We fix the parameter $\lambda \in (0,1)$ and two elements $y_1^*, y_2^* \in Y^*$. As the set $\partial G(x)$ is convex for any element $x \in dom\,G$, we may state:

$$\begin{aligned} &\inf_{x^*\in\partial G(x)} K^*(\,\Lambda^*(\,\lambda y_1^* + (1-\lambda)y_2^*\,) + x^*\,) \\ &\quad= \inf_{x_1^*\in\partial G(x)}\ \inf_{x_2^*\in\partial G(x)} K^*(\,\Lambda^*(\lambda y_1^* + (1-\lambda)y_2^*) + (\lambda x_1^* + (1-\lambda)x_2^*)\,) \\ &\quad\leq \inf_{x_1^*\in\partial G(x)}\ \inf_{x_2^*\in\partial G(x)}\Big\{\,\lambda\,K^*(\,\Lambda^* y_1^* + x_1^*\,) + (1-\lambda)\,K^*(\,\Lambda^* y_2^* + x_2^*\,)\Big\} \\ &\quad= \lambda \inf_{x_1^*\in\partial G(x)} K^*(\,\Lambda^* y_1^* + x_1^*\,) + (1-\lambda)\inf_{x_2^*\in\partial G(x)} K^*(\,\Lambda^* y_2^* + x_2^*\,) \ . \end{aligned}$$

Since convex functions, which can be bounded from above on some open set, are locally Lipschitz continuous in the interior of their effective domain (see e. g. [60] corollary 2.4 in chapter I), we conclude that the error majorant $\hat{M}_K(x,\cdot\,)$ itself is a locally Lipschitz continuous function. We are going to demonstrate, that $\hat{M}_K(x,\cdot\,)$ is also coercive, provided the following conditions are met: the functional G is continuous in a neighbourhood around the point $x \in dom\,G$ and F^* is coercive on the set $Y_0^* := \{\, y^* \in Y^* \mid \Lambda^* y^* = 0\,\}$. We note:

$$\begin{aligned} K^*(\Lambda^* y^* + x^*) &= \sup_{s\in X}\Big\{\,\langle s, \Lambda^* y^* + x^*\rangle_X - \inf_{y\in Y}\{\,H^{**}(\Lambda s - y) + h(y)\,\}\Big\} \\ &\geq \sup_{s\in X}\sup_{x\in X}\Big\{\,\langle\,(s-x)+x\,,\,\Lambda^* y^* + x^*\,\rangle_X - H^{**}(\Lambda s - \Lambda x) - h(\Lambda x)\Big\} \\ &\geq -H^{**}(0) + \sup_{x\in X}\Big\{\,\langle x, \Lambda^* y^* + x^*\rangle_X - h(\Lambda x)\Big\} \\ &\geq -H(0) + \sup_{r\geq 0}\sup_{\|x\|=1}\Big\{\,r\,\langle x, \Lambda^* y^* + x^*\rangle_X - h_0^{**}(\,r\,\|\Lambda x\|_Y)\Big\} \quad . \end{aligned}$$

The function h_0^{**} coincides with $h_0 \in \mathfrak{D}_0$ on the interval $[0, +\infty)$. Accordingly, we may exploit the continuity of the operator Λ and decrease the lower bound for K^* by substituting an upper bound for $\|\Lambda x\|_X$. We may postulate a number $\Lambda_0 > 0$ with the property:

$$\begin{aligned} K^*(\Lambda^* y^* + x^*) &\geq -H(0) + \sup_{r\geq 0}\sup_{\|x\|=1}\Big\{\,r\,\langle x, \Lambda^* y^* + x^*\rangle_X - h_0(\,r\,\Lambda_0\|x\|_X)\Big\} \\ &= -H(0) + \sup_{r\geq 0}\Big\{\,r\,\|\Lambda^* y^* + x^*\|_{X^*} - h_0(\,r\,\Lambda_0)\Big\} \\ &= -H(0) + h_0^*(\,\Lambda_0^{-1}\,\|\Lambda^* y^* + x^*\|_{X^*}) \quad . \end{aligned} \tag{1.33}$$

A closer inspection of the proofs in [141] shows, that the function h_0^* is monotonously increasing. Furthermore, it is known (see e. g. theorem 4.4 in [22]) that the subdifferential $\partial G(x)$ is contained within some ball of finite diameter centered at $0 \in X^*$, if and only if G is locally Lipschitz continuous around $x \in dom\, G$. In this case we infer from (1.33):

$$\inf_{x^* \in \partial G(x)} K^*(\Lambda^* y^* + x^*) \quad \longrightarrow \quad +\infty \qquad ; \qquad \|y^*\|_{Y^*/Y_o^*} \to +\infty \quad .$$

Since $M_F(\Lambda x, \cdot\,)$ is bounded from below by $H(0)$, we conclude that the error majorant $\hat{M}_K(x, \cdot\,)$ is coercive on the quotient space Y^*/Y_0^*. We assume, $\{y_j^* + \hat{y}_j^*\}_{j\in\mathbb{N}}$ is a minimising sequence, such that $y_j^* \in Y^*/Y_0^*$ and $\hat{y}_j^* \in Y_0^*$ hold for any index $j \in \mathbb{N}$. With a view to the above result the sequence $\{y_i^*\}_{j\in\mathbb{N}}$ must be bounded. Invoking the property vii listed in section 1.1.6 we find, that $\{F^*(-y_j^*)\}_{j\in\mathbb{N}}$ is bounded from above by some constant F_∞^*. We may state:

$$2\, F^*(\,\hat{y}_j^*/2\,) \;-\; F_\infty^* \quad \leq \quad 2\, F^*(\,\hat{y}_j^*/2\,) \;-\; F^*(\,-y_j^*\,) \quad \leq \quad F^*(\,y_j^* + \hat{y}_j^*\,)$$

since the functional F^* is convex. Consequently, we can control the first part of $\hat{M}_K$ by:

$$\begin{aligned} \hat{M}_F(\Lambda x, y_j^* + \hat{y}_j^*) \quad &\geq \quad H(0) \;+\; F(\Lambda x) \;+\; F^*(y_j^* + \hat{y}_j^*) \;-\; \langle\, \Lambda x, y_j^* + \hat{y}_j^* \,\rangle_Y \\ &\geq \quad H(0) \;+\; F(\Lambda x) \;-\; \langle\, \Lambda x, y_j^* \,\rangle_Y \;-\; F_\infty^* \;+\; 2\, F^*(\,\hat{y}_j^*/2\,) \quad . \end{aligned}$$

We conclude, that each minimising sequence of the error majorant $\hat{M}_K(x, \cdot\,)$ is bounded and features a weakly convergent subsequence. The limit of this subsequence is the minimiser of the error majorant due to the lower semicontinuity of $\hat{M}_K$ in the weak topology. However, the minimiser need not necessarily be unique, unless perchance $\hat{M}_K$ be strictly convex.

1.2.7 A review of our findings

We have derived not exactly two bounds on the energy error, we incur when we approximate the solution $x_0 \in X$ of the minimisation problem (1.5) by some function $x \in dom\, G$, but rather two families of error estimates that depend on a parameter $y^* \in Y^*$. Hence, finding a good - in some cases perhaps even finding a meaningful - estimate of the error may require the solution of a minimisation problem similar to (1.9) in terms of this parameter. In the introduction we have endeavoured to show that a number of established a posteriori error estimates for elliptic problems can be obtained from a complementary variational formulation, if we make special choices for the dual variables. Since we have rid ourselves from any admissibility constraints to be met by the dual variables, we can use the very same methods to construct the parameter y^*. In consequence, we may hope the generalised hypercycle estimates $M_K(x, \cdot\,)$ and $\hat{M}_K(x, \cdot\,)$ will fail only in those cases, in which the more conventional techniques will be found wanting as well.

We have seen in section 1.2.6, that our estimates can detect, whether we have found the analytical solution of the minimisation problem (1.5): in this sense the penalisation we have introduced in the preceding sections is at least consistent. How much the accuracy of the estimates suffers from the introduction of the expansion (1.10) and the subsequent separation of the dual and primal variables is difficult to assess. Under moderately restrictive assumptions on the algebraic manipulations we have introduced in section 1.2.4 the larger error bound $\hat{M}_K(x, \cdot\,)$ has been discovered to be locally Lipschitz continuous. Further conditions on the data of the problem (1.5) warrant, that the majorant is moreover coercive. Both features certainly aid us in minimising $\hat{M}_K$ with respect to its second argument. In the special case, that F is a quadratic form, the generalised hypercycle estimate $\hat{M}_K$ is indeed uniformly convex. Unfortunately, the smaller error majorant M_K seems much less amenable to our analysis, even if we assume F to be quadratic. While we can assert at least in this latter case, that a minimiser exists for the larger majorant $\hat{M}_K$, we have failed to obtain an existence result for M_K altogether. Let us summarise our findings in the following proposition:

Proposition 1.1 *Let X and Y denote two reflexive Banach spaces and $\Lambda : X \longrightarrow Y$ a continuous linear operator. The spaces X^* and Y^* are paired with X respectively Y. The dual pairings are designated $\langle \cdot, \cdot \rangle_X : X \times X^* \longrightarrow \mathbb{R}$ and $\langle \cdot, \cdot \rangle_Y : Y \times Y^* \longrightarrow \mathbb{R}$. The operator Λ is assumed coercive.*

Let $F : Y \longrightarrow \mathbb{R}$ designate an uniformly convex functional, while $G : X \longrightarrow (-\infty, +\infty]$ be proper and convex. The unique solution of the minimisation problem

$$F(\Lambda x) \;+\; G(x) \quad \underset{x \in X}{\longrightarrow} \quad \inf$$

be denoted by $x_0 \in X$. There is a forcing function $\phi \in \mathfrak{D}_0$, a set whose definition can be found in section 1.1.6, such that for any two elements $y, \hat{y} \in Y$

$$\phi(\,\|\,y - \hat{y}\,\|_Y) \quad \leq \quad F(y) \;+\; F(\hat{y}) \;-\; 2\,F\big(\tfrac{y+\hat{y}}{2}\big)$$

is met. Furthermore, a second function $f_0 \in \mathfrak{D}_0$ can be found, such that for any two elements $y, \hat{y} \in dom\,F$ and any functional $y^* \in \partial F(y)$ the following inequality holds:

$$F(\hat{y}) \quad \geq \quad F(y) \;+\; \langle\,\hat{y} - y, y^*\,\rangle_Y \;+\; f_0(\,\|\,\hat{y} - y\,\|_Y) \quad .$$

Let $H : Y \longrightarrow \mathbb{R}$ denote a proper function and $x_h \in dom\,G$ an arbitrary element understood to approximate the minimiser x_0. A family of functionals $K_y : X \longrightarrow \mathbb{R}_0^+$ is defined by:

$$K_y(s) \;:=\; \sup_{r \geq 0} \big\{\; r\,\|\,y + \Lambda s\,\|_Y \;-\; f_0^*(r)\;\big\} \quad ; \quad s \in X$$

for any element $y \in Y$. From these functionals another functional $K : X \longrightarrow \mathbb{R}$ is derived:

$$K(s) \;:=\; \inf_{y \in Y} \big\{\; H^{**}(-y) \;+\; K_y(s)\;\big\} \quad ; \quad s \in X \quad .$$

The distance between the minimiser x_0 and its approximation x_h can be measured in terms of the so called energy norm by the following abstract a posteriori estimate:

$$2\,\phi(\,\|\,\Lambda x_0 - \Lambda x_h\,\|_Y) \quad \leq \quad M_F(\Lambda x, y^*) \;+\; M_G(x, -\Lambda^* y^*) \quad ; \quad y^* \in Y^* \quad .$$

The nonnegative contributions M_F and M_G to the error bound are thereby defined as:

$$\begin{aligned} M_F(y, y^*) \;&:=\; F(y) \;+\; F^*(y^*) \;-\; \langle\,y, y^*\,\rangle_Y \quad , \\ M_G(x, x^*) \;&:=\; G(x) \;+\; G^*(x^*) \;-\; \langle\,x, x^*\,\rangle_Y \quad . \end{aligned}$$

While the first contribution M_F is finite for any choice of the arguments $y \in Y$ and $y^* \in Y^*$, the second contribution M_G may attain the value $+\infty$ for certain $x \in X$ and $x^* \in X^*$. To avoid the dual admissibility constraint $-\Lambda^* y^* \in dom\,G^*$, the following error bounds are specified:

$$M_K(x, y^*) \;:=\; M_F(\Lambda x, y^*) \;+\; \inf_{x^* \in \partial G(x)} K^*_{dF^*(y^*) - \Lambda x}(\Lambda^* y^* + x^*) \quad ,$$

$$\hat{M}_K(x, y^*) \;:=\; M_F(\Lambda x, y^*) \;+\; H(\,\Lambda x - dF^*(y^*)\,) \;+\; \inf_{x^* \in \partial G(x)} K^*(\Lambda^* y^* + x^*) \quad .$$

These bounds on the approximation error are reliable for any choice of the dual variable $y^* \in Y^*$:

$$2\,\phi(\,\|\,\Lambda x_0 - \Lambda x_h\,\|_Y) \quad \leq \quad M_K(x_h, y^*) \quad \leq \quad \hat{M}_K(x_h, y^*) \quad .$$

If the functional $H : Y \longrightarrow \mathbb{R}$ has the property $H(0) \leq H$ the above error bounds are sharp:

$$\inf_{y^* \in Y^*} M_K(x_0, y^*) \quad = \quad \inf_{y^* \in Y^*} \hat{M}_K(x_0, y^*) \quad = \quad 0 \quad .$$

If the functionals H and H^* are locally Lipschitz continuous, so is $\hat{M}_K(x, \cdot\,)$. If moreover G is continuous in a neighbourhood around the point $x \in dom\,G$ and if the functional F^* is coercive on the set $Y_0^* := \{\, y^* \in Y^* \,|\, \Lambda^* y^* = 0\,\}$, the generalised hypercycle estimate $\hat{M}_K(x, \cdot\,)$ is coercive on Y^*. In the special case, that F is a quadratic form, the error majorant $\hat{M}_K(x, \cdot\,)$ is uniformly convex, if the functional H is convex.

Proof Proof for all of the above statements has already been presented in section 1.2.6. □

In many an application both M_K and $\hat{M}_K$ can be impossible to evaluate immediately: in our introductory remarks we have mentioned the Dirichlet problem as an example. Since the generalised hypercycle estimate $\mathcal{H}^{(\kappa)}$ contains an expression defined but in a weak sense, we must find a bound on the dual norm using those techniques employed in the analysis of conventional a posteriori error estimators, which rely on the computation of residual expressions. Alternatively, we must assume, the dual variable exceeds its natural regularity requirements. In such a case we are able to bound the dual norm by invoking Friedrich's inequality:

$$| \nabla\cdot \sigma + f |_{\Omega,-1} \quad = \quad \sup_{v\in H^1_0(\Omega)} \frac{(\, v, f + \nabla\cdot \sigma \,)_\Omega}{| \, v \, |_{\Omega,1}} \quad \leq \quad \frac{1}{\lambda_0} \, \| \nabla\cdot \sigma + f \|_\Omega \quad ; \quad \sigma \in H_{\rm div}(\Omega) \ .$$

Conceptually, the above manipulation is the result our replacing the function $K : X \longrightarrow \mathbb{R}$ with a minorant $\tilde{K} : \tilde{X} \longrightarrow \mathbb{R}$, which is defined on a space larger than X. The associated dual space $\tilde{X}^*$ is smaller than X^*, whence we must ensure our choice of the dual parameter $y^* \in Y^*$ does not violate the requirement $\{\Lambda^* y^*\} + \partial G(x) \subset \tilde{X}^*$. In our introductory example this difficulty is reflected by the regularity condition $\sigma \in H_{\rm div}(\Omega)$. As a corollary of proposition 1.1 we state:

Proposition 1.2 *Let $\tilde{X} \supset X$ denote a reflexive Banach space and $\tilde{K} : \tilde{X} \longrightarrow \mathbb{R}$ a minorant for the functional $K : X \longrightarrow \mathbb{R}$ as defined by (1.19). It is assumed, that G can be extended to the space $\tilde{X}$ and that there is a subspace $\tilde{Y}^* \subset Y^*$ with the property $\Lambda^*(\tilde{Y}^*) \subset \tilde{X}^*$, which contains the solution y_0^* of the dual formulation. Furthermore, the functionals H and H^* are supposed to be continuous with $H(0) \leq H$. The quantity*

$$\tilde{M}_K(x, y^*) \ := \ M_F(\Lambda x, y^*) \ + \ H(\, \Lambda x - dF^*(y^*) \,) \ + \ \inf_{x^* \in \partial G(x)} \tilde{K}^*(\Lambda^* y^* + x^*)$$

defines an upper bound on the abstract a posteriori estimate $M_F(\Lambda x, y^*) + M_G(x, -\Lambda^* y^*)$ as specified by (1.8) for any choice of the dual parameter $y^* \in \tilde{Y}^*$. The majorant $\tilde{M}_K$ is continuous over the interior of its effective domain and coercive, if $\hat{M}_K(x, \cdot\,)$ is coercive. Let $x_0 \in X$ denote the solution of the problem (1.5). If $H(0) \leq \tilde{K}$ holds, the error majorant $\tilde{M}_K$ satisfies:

$$\inf_{y^* \in \tilde{Y}^*} \tilde{M}_K(x_0, y^*) \quad = \quad 0 \ .$$

Proof Since $\partial G(x)$ is a convex set, the infimum of $\tilde{K}^*(\Lambda^* y^* + x^*)$ with respect to $x^* \in \partial G(x)$ is a convex function in $y^* \in \tilde{Y}^*$. As such it is continuous over the interior of its effective domain (see e. g. corollary 2.5 in [60]). As the remaining constituents are continuous functions either by assumption or by virtue of the uniform convexity of F, the majorant $\tilde{M}_K$ is continuous over its effective domain. As K^* provides a lower bound on $\tilde{K}^*$, the majorant $\tilde{M}_K$ is coercive, whenever $\hat{M}_K$ is coercive. By assumption there is an element $y_0^* \in \tilde{Y}^*$ with the property:

$$-\Lambda^* y_0^* \ \in \ \partial G(x_0) \qquad \wedge \qquad y_0^* \ \in \ \partial F(\Lambda x_0) \ .$$

In consequence, the infimum can be from above bounded by $H(0) + \tilde{K}^*(0)$. We conclude:

$$\inf_{y^* \in \tilde{Y}^*} \tilde{M}_K(x_0, y^*) \quad \leq \quad H(0) \ + \ \sup_{s \in \tilde{X}} \{\, -\tilde{K}(s) \,\} \quad \leq \quad H(0) \ - \ H(0) \ .$$

□

Remark 1.1 The condition that the solution y_0^* of the dual formulation be contained in the subspace $\tilde{Y}^*$ may be viewed as a regularity requirement for the solution of the primal formulation (1.5) and hence as a condition to be met by the data of the problem. Let us revisit the Dirichlet problem and the estimate $\mathcal{H}^{(\kappa)}$: the solution y_0^* of the dual formulation reads ∇x_0, while $H_{\rm div}(\Omega)$ takes the role of $\tilde{Y}^*$ and $L^2(\Omega)$ that of $\tilde{X}$. We note, the requirement $\Lambda^*(\tilde{Y}^*) \subset \tilde{X}^*$ is satisfied. The second condition $y_0^* \in \tilde{Y}^*$ translates into $\Delta x_0 \in L^2(\Omega)$. Therefore, the right hand side must be contained at least in $L^2(\Omega)$. We conclude: $\partial G(x) = \{-f\} \subset \tilde{X}^*$ for any $x \in \tilde{X}$.

Remark 1.2 Both generalised hypercycle estimates M_K and $\hat{M}_K$ consist of two contributions, one of which is in a sense well behaved and one that is not. As we have already mentioned in the introductory remarks to this section, the sharper error majorant M_K seems almost impossible to analyse within our abstract framework. Its well behaved part $M_F(\Lambda x, \cdot\,)$ is convex and continuously differentiable with respect to its second argument. Apart from the fact, that stays bounded on bounded sets, we know next to nothing about its second part, however. The larger error estimate $\hat{M}_K$ is much simpler to analyse. The unproblematic part is convex, locally Lipschitz continuous and even coercive under reasonable assumptions on the functionals H and F^*. The other part is merely locally Lipschitz continuous, if the functional H is sufficiently regular. While the existence of a minimiser for the first part of the error majorant alone can be inferred from standard arguments, the continuous perturbation which is introduced by the term $H(\Lambda x - dF^*(y^*))$ prevents us from finding the graph of $\hat{M}_K$ weakly closed. It seems, that in [105] a result has been obtained, which warrants the existence of a minimiser for $\hat{M}_K$ nevertheless, if the generalised gradient of $\hat{M}_K$ in the sense of [51] meets certain conditions.

1.3 Using duality techniques to obtain bounds on functional outputs

In many applications from the physical sciences variational problems arise from the so called *Hamilton's principle*, that is a minimisation principle for the action integral. If finite element technology is used to approximate the solution of such variational problems, it seems therefore reasonable to use a posteriori error estimators of the kind discussed in section 1.2 in order to obtain adapted and possibly sparse computational meshes.

However, quite a number of situations may be imagined, in which the solution of some differential equation or inclusion is not the primary concern but rather a means of computing the relevant data. For instance, the solution of a flow problem around an airfoil may only serve to find the drag and the lift of said airfoil, whereas the exact calculation of the flow field is merely circumstantial. To give another example, let us consider a contact problem for an elastic workpiece. While it is necessary to solve the Lamé equations before the contact can be located, the deformation of the elastic body may be an irrelevant piece of information in itself. With a view to speeding up the calculations it may be desirable therefore to work with coarse grids and a low accuracy. But if the contour of the contact must be known accurately, how do we maintain the required resolution while thinning out the computational grids?

Obviously, this latter question is not easily addressed by any mathematical calculus that is derived from the duality arguments used in the previous section. In the following we will concern ourselves with a much more simple task: Provided there is a numerical approximation $x \in X$ to the solution $x_0 \in X$ of the variational problem

$$F(\Lambda x) \;+\; G(x) \qquad \underset{x \in X}{\longrightarrow} \qquad \inf$$

we will compute upper respectively lower bounds $M_+(\psi, x)$ and $M_-(\psi, x)$ to the output of some functional $\psi : X \longrightarrow \mathbb{R}$, which is applied to the solution x_0. We have to ask ourselves, for which kind of functionals ψ we will be able to find estimates of the form

$$M_-(\psi, x) \quad \leq \quad \psi(x_0) \quad \leq \quad M_+(\psi, x) \;\; .$$

As it will turn out, the calculation of a meaningful upper bound $M_+(\psi, x)$ will be impossible without the assumption that the functional ψ has a real valued concave hull. In estimating the lower bound $M_-(\psi, x)$ it will be necessary, however, to suppose, that $co\,\psi$ is real valued. Therefore, we can only expect satisfactory error estimates for a functional ψ, that is linear or behaves asymptotically almost like a linear functional.

1.3.1 Treatment of the linear case

Let us start our investigation by considering a very simple model problem. Thus we will gain the necessary insight into the matter at hand to proceed with our studies of the general case. We will assume, that X and Y are Hilbert spaces, such that we can identify X and X^* respectively Y

and Y^* with the help of the inner products $(\cdot,\cdot)_X$ and $(\cdot,\cdot)_Y$. Let $x_0 \in X$ denote the minimiser of the following quadratic functional:

$$J(x) \;=\; \frac{1}{2}\,(\,\Lambda\,x, \Lambda\,x\,)_X \;-\; (\,x, f\,)_X \qquad ; \qquad x \in X$$

with some operator $\Lambda : X \longrightarrow Y$ as specified in subsection 1.2.1 and some function $f \in X$. Said minimiser $x_0 \in X$ may be exhibited as the solution of the following variational formulation:

$$(\,\Lambda\,\xi, \Lambda\,x_0\,)_Y \;=\; (\,\xi, f\,)_X \qquad ; \qquad \xi \in X \quad .$$

Let us suppose furthermore, we want to obtain bounds on the output of the linear functional

$$\psi(x) \;:=\; (\,x, \psi\,)_X \qquad ; \qquad x \in X$$

for some arbitrary element $\psi \in X$. We may conceive the following saddle point problem:

$$\psi(x_0) \;=\; \sup_{x\in X}\,\inf_{\xi\in X}\, L(x,\xi) \;:=\; \sup_{x\in X}\,\inf_{\xi\in X}\,\big\{\,(\,x,\psi\,)_X \,-\, (\,\Lambda\xi, \Lambda x\,)_Y \,+\, (\,\xi, f\,)_X\,\big\}\;.$$

Its formulation is in fact trivial as a finite infimum of the Lagrangian $L(x,\xi)$ with respect to the second argument $\xi \in X$ can only be attained if $x = x_0$ holds:

$$\inf_{\xi\in X}\, L(x,\xi) \;=\; \begin{cases} (\,x,\psi\,)_X & ; \quad x = x_0 \\ -\infty & ; \quad \text{else} \end{cases} \qquad . \tag{1.34}$$

An upper bound for the functional output $\psi(x_0)$ may be found by invoking the dual formulation:

$$\check{M}_+(\psi,\xi) \;:=\; \sup_{x\in X}\, L(x,\xi) \;=\; \begin{cases} (\,\xi, f\,)_X & ; \quad \xi = \xi_0 \\ +\infty & ; \quad \text{else} \end{cases} \qquad .$$

Hereby, the element $\xi_0 \in X$ designates the solution of the so called *dual problem*

$$(\,\Lambda\,\xi_0, \Lambda\,x\,)_Y \;=\; (\,x, \psi\,)_X \qquad ; \qquad x \in X \quad ,$$

if we use the terminology introduced in [20]. We conclude:

$$\psi(x_0) \;\leq\; \inf_{\xi\in X}\, \check{M}_+(\psi,\xi) \;=\; (\,\xi_0, f\,)_X \quad . \tag{1.35}$$

Since we will be unable in general to determine the solution ξ_0 of the dual problem exactly, the a posteriori error estimate (1.35) is useless from a mathematical point of view. Any numerical approximation $\xi \in X$ to this solution will cause the error bound $M_+(\psi,\xi)$ to become infinite due to the linear dependency of the Lagrangian L on its first argument. Of course, we may still evaluate the expression $(\xi, f)_X$. We cannot warrant, however, that the result is an upper bound on $\psi(x_0)$. We may lift this difficulty by *augmenting* the Lagrangian L with a quadratic term, which disappears at the very point x_0. Thus we obtain:

$$\tilde{L}(x,\xi) \;:=\; L(x,\xi) \;-\; (\,\Lambda\,x, \Lambda\,x\,)_Y \;+\; (\,x, f\,)_X \qquad ; \qquad x,\,\xi \,\in\, X \quad .$$

Obviously, we can still recover the output of the functional ψ by first taking the infimum of the augmented Lagrangian $\tilde{L}$ with respect to its second argument $\xi \in X$. But now we find:

$$\tilde{M}_+(\psi,\xi) \;:=\; \sup_{x\in X}\,\tilde{L}(x,\xi) \;=\; (\,\xi, f\,)_X + \sup_{x\in X}\big\{\,(x, f+\psi)_X - (\,\Lambda\,(x+\xi), \Lambda\,x\,)_Y\,\big\} \;<\; +\infty$$

independently of our choice of the dual variable $\xi \in X$. For all practical purposes, we cannot evaluate the above error estimate, since its computation would entail the analytical treatment of a maximisation task, that is as difficult to solve as the original minimisation problem for the functional J. We would like to remark, that the a posteriori error estimator proposed in [116] suffers from this very same drawback. (The formula 35 on page 206 contains two errors, whence the problem is not immediately apparent.)

In [116] the difficulty of finding the exact solution of some suitably constructed auxiliary problem is resolved by avoiding the continuous formulation altogether. Instead of estimating the expression $\psi(x_0)$ two finite dimensional trial spaces $X_H \subset X_h$ are used to bound the quantity $\psi(x_h)$ with $x_h \in X \cap X_h$ being an approximation of the true solution x_0. To keep the numerical complexity of the algorithm low, hybridisation is employed for the X_H discretisation, such that the auxiliary problem derived from the X_h discretisation decouples into independent, locally defined subproblems. In [20, 119–121] a device known as *Galerkin orthogonality* is exploited instead. Assuming $X_h \subset X$ we infer from (1.35):

$$\psi(x_0 - x_h) \;\leq\; (\,\xi_0, f\,)_X \;-\; (\,\Lambda\,\xi_0, \Lambda\,x_h\,)_Y \;=\; (\,\xi_0 - \xi_h, f - \Lambda^*\Lambda\,x_h\,)_X$$

with $x_h \in X_h$ designating the numerical solution of the original variational problem and $\xi_h \in X_h$ an arbitrary test function. Choosing ξ_h judiciously we can exploit the interpolation properties of the trial space X_h and afterwards invoke a priori estimates for the dual problem in order to bound the higher order derivatives of ξ_0. Clearly, the procedure suffers from the fact, that neither interpolation nor stability constants are readily available. Furthermore, its applicability is limited to the linear and unconstrained case.

We will address the difficulties inherent to the evaluation of $\tilde{M}_+(\psi, \xi)$ with the help of a duality argument. Though we will not be able to apply the results of section 1.2 directly to the resulting estimate we will still find the analytical technology very useful, that has been developed in that section. We note, that the abstract error estimate, we have derived in subsection 1.2.2, may serve the very same purpose of augmenting the Lagrangian L as the quadratic expression the have employed above. Indeed, both expressions are closely related:

$$(\,\Lambda\,x, \Lambda\,x\,)_Y \;-\; (\,x, f\,)_X \;+\; G^*(-\Lambda^*\Lambda\,x) \;=\; M_F(\Lambda\,x, \Lambda\,x) \;+\; M_G(x, -\Lambda^*\Lambda\,x) \;\;,$$

if we introduce the following notation: $F(y) := 1/2\,(y, y)_Y$ and $G(x) := (-x, f)_X$ for all elements $y \in Y$ and $x \in X$. At the very point x_0 the last term on the left hand side vanishes:

$$G^*(x) \;=\; \begin{cases} 0 & ; \quad x = -f \\ +\infty & ; \quad \text{else} \end{cases} \;\;,$$

for $\Lambda^*\Lambda\,x_0 = f$ is the necessary optimality condition of our model problem. Let us define:

$$L(x, y, \xi) \;:=\; L(x, \xi) \;-\; F(\Lambda\,x) \;-\; F^*(y) \;-\; G(x) \;-\; G^*(-\Lambda^* y) \quad ; \quad x, \xi \in X,\, y \in Y \;\;.$$

With a view to (1.8) and (1.34) it is obvious, that this new Lagrangian has the property:

$$\psi(x_0) \;=\; \sup_{y \in Y} \left\{\, (\,x_0, \psi\,)_X \,-\, M_F(\Lambda x_0, y) \,-\, M_G(x_0, -\Lambda^* y) \,\right\} \;=\; \sup_{x \in X} \sup_{y \in Y} \inf_{\xi \in X} L(x, y, \xi) \;\;.$$

Hence, we can introduce the following upper bound on the functional output $\psi(x_0)$:

$$\hat{M}_+(\psi, \xi) \;:=\; (\,\xi, f\,)_X \,-\, \inf_{x \in X} \inf_{y \in Y} \left\{\, F(\Lambda x) + F^*(y) + G^*(-\Lambda^* y) + (\,\Lambda\,\xi, \Lambda\,x\,)_Y - (\,x, f + \psi\,)_X \right\}$$

for any $\xi \in X$. We will now consider the dual formulation in order to obtain an upper bound of this last expression. In preparation of this step let us define the following affine space:

$$Q_\xi \;:=\; \left\{\, u \in Y \;\middle|\; (\,\Lambda x, u\,)_Y + (\,\Lambda x, \Lambda\,\xi\,)_Y \,=\, (\,x, f + \psi\,)_X \;;\; x \in X \,\right\} \;\;, \tag{1.36}$$

for we are going to face the very same predicament we have encountered in dealing with a posteriori error estimates for the energy norm: the dual estimates need not necessarily be finite, unless certain constraints on the dual variables have been met. We note:

$$\begin{aligned} \hat{M}_+(\psi, \xi) \;&=\; (\,\xi, f\,)_X \;-\; \inf_{x \in X} \inf_{y \in Y} \left\{ \sup_{u \in Y} \left\{\, (\,\Lambda x, u\,)_Y \,-\, F^*(u) \,\right\} \,+\, \sup_{v \in Y} \left\{\, (\,v, y\,)_Y \,-\, F(v) \,\right\} \right. \\ &\qquad \left. +\; \sup_{s \in X} \left\{\, (\,s, -\Lambda^* y\,)_Y \,-\, G(s) \,\right\} \,+\, (\,\Lambda\,\xi, \Lambda\,x\,)_Y \,-\, (\,x, f + \psi\,)_X \right\} \\ &\leq\; (\,\xi, f\,)_X \;-\; \sup_{u \in Y} \sup_{v \in Y} \sup_{s \in X} \left\{ \inf_{x \in X} \inf_{y \in Y} \left\{\, (\,\Lambda x, u\,)_Y + (\,v, y\,)_Y - (\,s, \Lambda^* y\,)_X \,+ \right.\right. \end{aligned}$$

$$+ \ (\Lambda\,\xi, \Lambda\,x\,)_{_Y} - \ (x, f + \psi\,)_{_X} \ \Big\} \ - F^*(u) - F(v) - G(s) \ \Big\}$$

$$= \ (\xi, f\,)_{_X} \ + \ \inf_{s\in X} \inf_{u\in Q_\xi} \Big\{ \ F(\Lambda s) \ + \ G(s) \ + \ F^*(u) \ \Big\} \ . \tag{1.37}$$

The error bounds $\tilde{M}_+(\psi,\xi)$ and $\hat{M}_+(\psi,\xi)$ differ in fact but very little. While the evaluation of $\tilde{M}_+(\psi,\xi)$ necessitates our solving a maximisation problem in terms of the primal variable $x \in X$, the computation of (1.37) requires us to determine the set Q_ξ, whose elements are defined as solutions of a related variational formulation. The expression $F^*(u)$ can be viewed as a penalty term, which effectually removes from the minimiser $u_0 \in Y$ all those components, that would otherwise be contained in the kernel of Λ^*. Consequently, the solution $\tilde{x} \in X$ of the maximisation task associated with the evaluation of $\tilde{M}_+(\psi,\xi)$ is in some sense recovered: $u_0 = \Lambda\tilde{x}$.

Though our exposition may seem somewhat circumlocutory, it has lead to an error estimate we can easily put into a more general form than the one we have derived previously. Moreover, is now apparent, how we have to apply the analytical technology developed in section 1.2. As the estimate (1.37) becomes meaningless, if the condition $u \in Q_\xi$ is violated, we have to relax this requirement. Exploiting the properties of uniformly convex functions as they have been outlined in paragraph 1.1.6 we may postulate a convex function $h_0 \in \mathfrak{D}_0$ such that:

$$F^*(u) \ \leq \ F^*(y) \ + \ \langle\, dF^*(y), u - y \,\rangle_{_Y} \ + \ h_0^*(\|u - y\|_{_Y}) \qquad ; \qquad y \in dom\, F \ .$$

In the present instance we may turn this estimate into an equality, when we define the function h_0 by stipulating $h_0^*(t) := 1/2\,t^2$ for any $t \geq 0$. Employing the above result we find:

$$\begin{aligned} F(\Lambda s) \ + \ G(s) \ + \ F^*(u) \ &\leq \ F(\Lambda s) \ + \ F^*(y) \ + \ \langle\, dF^*(y), u - y \,\rangle_{_Y} \ + \ h_0^*(\ \|u - y\|_{_Y}) \\ &+ \ G(s) \ - \ \langle\, \Lambda s, y \,\rangle_{_Y} \ + \ \langle\, \Lambda s, u \,\rangle_{_Y} \ - \ \langle\, \Lambda s, u - y \,\rangle_{_Y} \\ &\leq \ M_F(\Lambda s, y) \ + \ \langle\, dF^*(y) - \Lambda s, u - y \,\rangle_{_Y} \ + \ G(s) \\ &+ \ h_0^*(\ \|y - u\|_{_Y}) \ + \ \langle\, \Lambda s, u \,\rangle_{_Y} \qquad ; \qquad y \in Y \quad . \end{aligned}$$

Again, we may use the device (1.14) in order to separate those contributions that correspond to the necessary optimality condition for the functional J from those contributions that are related to the relaxation of the constraint $u \in Q_\xi$. Invoking the definition (1.11) we thus obtain:

$$\begin{aligned} F(\Lambda s) \ + \ G(s) \ + \ F^*(u) \ &\leq \ M_F(\Lambda s, y) \ + \ H(\ \Lambda s - dF^*(y)\,) \ + \ H^*(y - u) \\ &+ \ G(s) \ + \ h^*(y - u) \ + \ \langle\, \Lambda s, u \,\rangle_{_Y} \qquad ; \qquad y \in Y \quad . \end{aligned}$$

We may now proceed exactly as we have done in section 1.2.4 and rid ourselves of the admissibility constraint $u \in Q_\xi$ by the introduction of a saddle point problem:

$$\inf_{u\in Q_\xi} \Big\{ \ H^*(u - y) \ + \ G(s) \ + \ h^*(u - y) \ + \ \langle\, \Lambda s, u \,\rangle_{_Y} \ \Big\} \quad =$$

$$= \ \inf_{u\in Y} \Big\{ \ \sup_{t\in X} \big\{ \ (\Lambda\, t, u\,)_{_Y} + \ (\Lambda\, t, \Lambda\, \xi\,)_{_Y} - \ (t, f + \psi\,)_{_X} \ \big\} \ + \ \langle\, \Lambda s, u \,\rangle_{_Y} \ + \ G(s)$$

$$+ \ \sup_{w\in Y} \big\{ \ (w, y - u\,)_{_Y} - \ H^{**}(w) \ \big\} \ + \ \sup_{z\in Y} \big\{ \ (z, y - u\,)_{_Y} - \ h(z) \ \big\} \ \Big\}$$

$$= \ \sup_{w\in Y} \sup_{z\in Y} \sup_{t\in X} \Big\{ \ \inf_{u\in Y} \big\{ \ (\Lambda\, t, u\,)_{_Y} - \ (w + z, u\,)_{_Y} + \ (\Lambda s, u\,)_{_Y} \ \big\} \ + \ G(s)$$

$$+ \ (\Lambda\, t, \Lambda\, \xi\,)_{_Y} - \ (t, f + \psi\,)_{_X} + \ (w + z, y\,)_{_Y} - \ H^{**}(w) \ - \ h(z) \ \Big\}$$

$$= \ \sup_{t\in X} \Big\{ \ (\Lambda\, t, \Lambda\, \xi\,)_{_Y} - \ (t, f + \psi\,)_{_X} + \ (\Lambda(t + s), y\,)_{_Y} + \ G(s) \ - \ K(t + s) \ \Big\}$$

$$= \ (s, \psi - \Lambda^*\Lambda\, \xi\,)_{_X} \ + \ K^*(\ \Lambda^*\Lambda\, \xi + \Lambda^* y - f - \psi\,) \qquad .$$

Hereby, we have used the very same function K we have introduced in (1.19). The existence of a saddle point, which in the present context is a well known fact, is not necessary to ensure that the above equation is also valid in a more general setting: We refer to proposition 2.3 in chapter VI of [60] for a proof. By combining our findings we now obtain:

$$\hat{M}_+(\psi,\xi) \;\leq\; \inf_{s\in X}\,\inf_{y\in Y}\Big\{\, M_F(\Lambda s, y) \,+\, H(\,\Lambda s - dF^*(y)\,) \,+\, (\,\xi, f\,)_X \tag{1.38}$$

$$+\;\; (\,s, \psi - \Lambda^*\Lambda\,\xi\,)_X \,+\, K^*(\,\Lambda^*\Lambda\,\xi + \Lambda^* y - f - \psi\,)\,\Big\} \quad .$$

The a posteriori estimate (1.38) yields a computable upper bound on the functional output $\psi(x_0)$, if we assume that the function H meets those requirements imposed in section 1.2.6. In this case we can assert, that the right hand side stays finite for any choice of the parameters $s, \xi \in X$ and $y \in Y$. A lower bound on the output of ψ can be derived immediately by applying the above analysis to the functional $-\psi$. Thus we may conclude:

$$-\,M_+(-\psi,\xi,s,y) \;\leq\; \psi(x_0) \;\leq\; M_+(\psi,\xi,s,y) \;\;,$$

whereby $M_+(\psi,\xi,s,y)$ is defined by the right hand side of (1.38).

1.3.2 On possible extensions

In the treatment of a posteriori estimates for the bounds on functional outputs we have assumed, that both the objective functional and the necessary optimality condition, which determines the argument of that functional, be linear. Thus we have caused the functional output to depend on the data of the problem in a linear fashion. Consequently, the representation theorem of Riesz has warranted the existence of some function $\xi_0 \in X$ with the property: $\psi(x_0) = (\,\xi_0, f\,)_X$. We have demonstrated, that the function ξ_0 is related to the objective functional ψ by: $\Lambda^*\Lambda\xi_0 = \psi$. Hence, we may view the a posteriori estimate of the functional output defined by

$$M_+(\psi,\xi,s,y) \;:=\; (\,\xi, f\,)_X + M_F(\Lambda s, y) \,+\, H(\,\Lambda s - dF^*(y)\,)$$

$$+\;\; (\,s, \psi - \Lambda^*\Lambda\,\xi\,)_X \,+\, K^*(\,\Lambda^*\Lambda\,\xi + \Lambda^* y - f - \psi\,)$$

for any $\psi, \xi, s \in X$ and $y \in Y$ as an embodiment of the mapping $f \to \psi(x_0)$. Thereby, residual terms have been added to account for the function ξ_0 can not being computed exactly. We note:

$$\psi(x_0) \;=\; M_+(\psi, (\Lambda^*\Lambda)^{-1}\psi, x_0, \Lambda x_0) \qquad ; \qquad \psi \in X \quad .$$

Unfortunately, we will face a number of difficulties if we try and extend the theory of a posteriori estimates for functional outputs as it has been developed in the previous section to the more general case of uniformly convex variational problems. For one thing, additional complexities will arise from the necessity to consider two necessary optimality conditions of the form (1.12) instead of merely one such condition. Let us assume, that $\psi : Y \longmapsto \mathbb{R}$ is some functional, not necessarily linear, and $x_0 \in X$ the solution of the variational problem introduced in paragraph 1.2.1. Any solution $y_0^* \in Y^*$ of the associated dual formulation

$$F^*(-y^*) \,+\, G^*(\Lambda^* y^*) \qquad \underset{y^*\in Y^*}{\longrightarrow} \qquad \inf$$

can be characterised in the following fashion:

$$-\,\Lambda^* y_0^* \;\in\; \partial G(x_0) \qquad \wedge \qquad y_0^* \;\in\; \partial F(\Lambda x_0) \;\;.$$

With a view to the previous section these optimality conditions may prompt us to consider an augmented Lagrangian $L : X \times X^* \times Y^* \times Y^* \times X \longmapsto \mathbb{R}$, which is defined by:

$$L(x,x^*,y^*,\lambda^*,\mu) \;:=\; \psi(x) \;-\; \langle\, dF^*(y^*) - \Lambda\, x\,,\, \lambda^*\,\rangle_Y \;-\; \langle\, \mu\,,\, \Lambda^* y^* + x^*\,\rangle_X$$

$$-\quad F(\Lambda x) \;-\; G(x) \;-\; F^*(y^*) \;-\; G^*(-\Lambda^* y^*) \quad . \tag{1.39}$$

Hereby, we need to assume that the element $x^* \in X^*$ be contained in the subdifferential $\partial G(x)$ in order to account for the first of the above optimality conditions. An application of a duality argument would then result in an upper bound on $\psi(x_0)$ of the following form:

$$K(\mu, \lambda^*) \quad := \quad - \inf_{x \in dom\, G} \inf_{x^* \in \partial G(x)} \inf_{y^* \in Y^*} \left\{ - L(x, x^*, y^*, \lambda^*, \mu) \right\}$$

with $\mu \in X$ and $\lambda^* \in Y^*$. To find a reliable estimate of this last expression within the framework of conjugate duality we would have to apply the Fenchel transform to two families of functions:

$$g_\mu(x) \quad := \quad \inf_{x^* \in \partial G(x)} \langle\, \mu, x^* \,\rangle_X \qquad ; \qquad x \in dom\, G$$

$$f_{\lambda^*}(y^*) \quad := \quad \langle\, dF^*(y^*), \lambda^* \,\rangle_Y \qquad ; \qquad y^* \in Y^*$$

for any $\mu \in X$ and $\lambda^* \in Y^*$. However, by fixing these parameters we need not necessarily obtain convex functions. In fact, both g_μ and f_{λ^*} will be neither convex nor concave in general, as we can easily see, if we study for example real valued functionals of the form

$$F(u) \quad := \quad \frac{1}{p} \int_0^1 |u(x)|^p \; dx \qquad ; \qquad u \in L^p([0,1])$$

for some $p > 2$. Defining the dual exponent $q \in (1,2)$ in the usual manner by $p^{-1} + q^{-1} = 1$ we can compute the function f_{λ^*} for any given Lagrange multiplier $\lambda^* \in L^q([0,1])$ by:

$$f_{\lambda^*}(y^*) \quad = \quad \int_0^1 \frac{y^*(x)\, \lambda^*(x)}{|y^*(x)|^{2-q}} \; dx \qquad ; \qquad y^* \in L^q([0,1]) \;\; .$$

This function is neither convex nor concave, if $\lambda^* \neq 0$ holds. In that case its Fenchel transform in the convex sense attains the value $+\infty$ for any argument $y \in L^p([0,1])$, whence the biconjugate $f^{**}_{\lambda^*}$ supplies but a trivial lower bound on f_{λ^*}. We infer, that we need to ensure the convexity of the directional derivative f_{λ^*} in order to obtain a meaningful bound on $K(\mu, \lambda^*)$.

The above example challenges our mode of approach to finding a reliable estimate for the functional output $\psi(x_0)$. Hence, if we want to retain the saddle point formulation (1.39) we must obviate the difficulties that arise out of the lacking convexity of f_{λ^*} and mutatis mutandis g_μ. A conceivable attempt at solving this problem consists in replacing the functional $F : Y \longrightarrow \mathbb{R}$ with a quadratic approximation $Q_F : Y \longrightarrow \mathbb{R}$, such that $Q_F(y) \approx F(y)$ holds in some neighbourhood around the point Λx. Similarly, the functional $G : X \longrightarrow (-\infty, +\infty]$ is to be replaced by some other quadratic model, which approximates G in a neighbourhood of the numerical solution x. Subsequently, the results detailed in section 1.3.1 me be applied.

The difficulties we encounter when we embrace such an approach are obvious: We have to produce quadratic forms Q_F and Q_G, that approximate in some sense to be specified the given convex functionals F and G. Since we cannot know the solution x_0 of the variational problem (1.5) exactly, the construction of these forms must necessarily proceed in a haphazard manner. Assuming that the numerical approximation x is close to the solution x_0 and that both F and G are sufficiently smooth, we may try and expand these functionals into power series around the point x. Truncating these series behind the quadratic term is equivalent to a linearisation of the necessary optimality conditions around the numerical solution of the variational problem. As we abolish the original formulation we can no longer warrant, that our estimates of the functional output are reliable. Moreover, we won't even be able to assert, whether our results pertain to the original statement of the problem at all.

Chapter 2

Applications of error majorants derived from duality arguments

In the previous chapter we have detailed a theoretical framework for the computation of reliable bounds on the approximation error we incur in the numerical solution of convex variational problems. We have been able to adapt out analytical techniques in order to calculate reliable bounds on the output of a linear functional which is applied to the approximate minimiser of a quadratic form. In the following we will elaborate a number of important applications.

2.1 The Laplace Problem

The most generic example by which elliptic differential operators may be studied is certainly the Laplace problem. Therefore, we will start this chapter with an examination of duality error estimates for the approximation errors of numerical solutions which are measured in the Dirichlet norm. To underscore the importance of those modifications by which we have obtained reliable error bounds from the abstract duality estimates discussed in section 1.2.2 we commence our study with a comparison of the Laplace and of the Helmholtz problem. In the case of the latter problem we are able to evaluate the right hand side of (1.8) explicitly, while in the case of the seemingly simpler Laplace problem the more careful analysis we have rendered in the sections 1.2.4 and 1.2.5 is inevitable.

In the first section we introduce the function spaces we must employ to define the variational setting of the Laplace problem along with the appropriate notation. In the section 2.1.2 we apply the abstract results we have developed throughout chapter 1 to the Helmholtz problem. In the following section the Laplace problem is considered. Two families of a posteriori estimates are derived, whose properties are studied in the section 2.1.4. In the section 2.1.5 we discuss the relationship between these error bounds and the hypercycle estimates for the Helmholtz problem. Our findings are summarised in the section 2.1.6.

2.1.1 Some remarks on the notation

Let us suppose that $\Omega \subset \mathbb{R}^n$ with $n \in \{2, 3\}$ is a bounded domain with a smooth boundary. For any $\Omega' \subseteq \Omega$ the space of all p-summable functions with p-summable generalised derivatives up to order m and values in $\mathbb{R}^n$ shall be denoted $W^{m,p}(\Omega', \mathbb{R}^n)$. We can turn $W^{m,p}(\Omega', \mathbb{R}^n)$ into a Banach space (see e. g. §5.4 in [97]) if we introduce the norm

$$\| v \|^p_{p,\Omega',m} := \int\limits_{\Omega'} \sum_{i=1}^{n} \sum_{|\alpha| \leq m} \left| \frac{\partial^\alpha v_i}{\partial x^\alpha} \right|^p \, dx \quad ; \quad v \in W^{m,p}(\Omega', \mathbb{R}^n) \; . \tag{2.1}$$

The corresponding Banach space of scalar functions is abbreviated by $W^{m,p}(\Omega')$. By taking the closure of $C_0^\infty(\Omega', \mathbb{R}^n)$ in the above norm the subspace $W_0^{m,p}(\Omega', \mathbb{R}^n) \subset W^{m,p}(\Omega', \mathbb{R}^n)$ is obtained, whose elements have no trace on the boundary $\partial\Omega'$. For $m = 0$ both of these spaces coincide with the space $L^p(\Omega', \mathbb{R}^n)$ of all p-summable functions. In this case the first subscript of the norm (2.1) will be dropped. The associated seminorm is defined by:

$$| v |^p_{p,\Omega',m} := \int\limits_{\Omega'} \sum_{i=1}^{n} \sum_{|\alpha| = m} \left| \frac{\partial^\alpha v_i}{\partial x^\alpha} \right|^p \, dx \quad ; \quad v \in W^{m,p}(\Omega', \mathbb{R}^n) \; .$$

For any index $p \geq 2$ let us define the dual index $q \in (1, 2)$ by the formula $p^{-1} + q^{-1} = 1$. We may now introduce the space $W^{-m,q}(\Omega', \mathbb{R}^n)$ of all those vector valued distributions, such that

$$\langle v, v^* \rangle_{\Omega'} := \int\limits_{\Omega'} \sum_{i=1}^{n} v_i \, v_i^* \, dx \tag{2.2}$$

constitutes a continuous bilinear form for any $v \in W^{m,p}(\Omega', \mathbb{R}^n)$ and any $v^* \in W^{-m,q}(\Omega', \mathbb{R}^n)$. In the scalar case we will again drop the reference to that space, in which both functions and distributions attain their respective point values. For full particulars on the proper construction of dual linear spaces to Sobolev spaces of the type $W^{m,p}(\Omega', \mathbb{R}^n)$ we refer to [97]. A norm that is compatible with the dual pairing we have introduced by (2.2) may be defined by:

$$| u^* |_{q,\Omega,-m} \quad := \quad \sup_{u \in W_0^{m,p}(\Omega', \mathbb{R}^n)} \frac{\langle u, u^* \rangle_{\Omega'}}{| u |_{p,\Omega,m}} \qquad ; \qquad u^* \in W^{-m,q}(\Omega', \mathbb{R}^n) \; .$$

Finally, we will need the Banach space $W^{p,m}_{\mathrm{div}}(\Omega')$ of p-summable vector fields, whose divergence is also p-summable. This space can be obtained by taking the closure of $C^\infty(\Omega', \mathbb{R}^n)$ in the norm

$$\| \tau \|^p_{p,\Omega',m} \quad := \quad \| \tau \|^p_{p,\Omega',m} \; + \; \| \nabla \cdot \tau \|^p_{p,\Omega',m} \qquad ; \qquad \tau \; \in \; W^{m,p}_{\mathrm{div}}(\Omega') \; .$$

To simplify the notation we will suppress the first index, whenever $p = 2$ holds. To place emphasis on the Hilbert space setting we will replace the symbol W by the symbol H. In the following we will therefore write $H^m_{\mathrm{div}}(\Omega')$ instead of $W^{m,2}_{\mathrm{div}}(\Omega')$ and $H^m(\Omega', \mathbb{R}^n)$ in place of $W^{m,2}(\Omega', \mathbb{R}^n)$.

2.1.2 A duality estimate for the Helmholtz problem

Sticking to the notation we have used throughout the previous chapter let us introduce the following functionals with $f \in L^2(\Omega)$ and λ representing some positive parameter:

$$F(y) \quad := \qquad \frac{1}{2} \int\limits_\Omega \sum_{i=1}^n y_i(\xi)^2 \; d\,\xi \qquad ; \qquad y \in L^2(\Omega, \mathbb{R}^n)$$

$$G(x) \quad := \quad \frac{1}{2} \int\limits_\Omega \Big\{ \; \lambda^2 x(\xi)^2 \; - \; 2\, f(\xi)\, x(\xi) \; \Big\} \; d\,\xi \qquad ; \qquad x \in L^2(\Omega) \; .$$

The operator $\Lambda : X \longrightarrow Y$ shall be defined as the gradient mapping $\nabla : H^1(\Omega) \longrightarrow L^2(\Omega, \mathbb{R}^n)$. With these abbreviations we can rewrite the well known Helmholtz problem with homogeneous Dirichlet boundary conditions in the form (1.5), if we choose $X = H^1_0(\Omega)$. The conjugate functionals F^* and G^* are now obtained by invoking Lebesgue's theorem on dominated convergence. To give an example of this procedure let us elaborate the computation of $G^* : L^2(\Omega) \longrightarrow \mathbb{R}$:

$$G^*(x^*) \quad = \quad \sup_{x \in L^2(\Omega)} \Big\{ \; \langle x, x^* \rangle_\Omega \; - \; G(x) \; \Big\} \quad = \quad \sup_{x \in L^2(\Omega)} \int\limits_\Omega \Big\{ \; x\, x^* \; - \; \frac{\lambda^2 x^2}{2} \; + \; f\, x \; \Big\} \; d\,\xi$$

$$= \quad \int\limits_\Omega \sup_{t \in \mathbb{R}} \Big\{ \; t\, x^*(\xi) \; - \; \frac{\lambda^2\, t^2}{2} \; + \; f(\xi)\, t \; \Big\} \; d\,\xi \quad = \quad \frac{1}{2\,\lambda^2} \, \| \, x^* + f \, \|^2_\Omega \tag{2.3}$$

for any $x^* \in L^2(\Omega)$. Similarly, we find $F^*(\sigma) = 1/2 \, \|\sigma\|^2_\Omega$ for any vector field $\sigma \in L^2(\Omega, \mathbb{R}^n)$. We must be aware, however, that our results will depend on the domain of definition we assign to the functional G. If the data of the problem were less regular, that is $f \in H^{-1}(\Omega)$, we would have to define the functional G on the set $H^1(\Omega)$. Consequently, we would have to determine a conjugate G^*, that could act on any distribution $x^* \in H^{-1}(\Omega)$. We would no longer be able to derive a closed analytical expression for such a functional. By virtue of (1.8) we may restrict ourselves to an arbitrary subset of $L^2(\Omega, \mathbb{R}^n)$ in order to derive a bound on the approximation error. Hence, we may choose $\sigma \in H_{\mathrm{div}}(\Omega)$ to ensure we can evaluate the right hand side of (1.8) even if the conjugate functional G^* is used as it has been derived in (2.3). In the present instance the forcing function $\phi \in \mathfrak{D}_0$ may be defined by $\phi(t) = 1/4\, t^2$. Hence, we infer from (1.8):

$$1/2 \; | \, x - x_0 \, |^2_{\Omega,1} \quad \leq \quad F(\nabla x) \; + \; F^*(\sigma) \; + \; G(x) \; + \; G^*(\nabla \cdot \sigma) \qquad ; \qquad \sigma \in H_{\mathrm{div}}(\Omega) \; ,$$

whereby the function $x \in H^1_0(\Omega)$ is supposed to approximate the analytical solution $x_0 \in H^1_0(\Omega)$ of the Helmholtz problem. Combining the above formula with (2.3) we find:

$$| \, x - x_0 \, |^2_{\Omega,1} \quad \leq \quad \| \, \nabla x \, \|^2_\Omega \; + \; \| \, \sigma \, \|^2_\Omega \; + \; \lambda^2 \, \| \, x \, \|^2_\Omega \; - \; 2 \, \langle \, x, f \, \rangle_\Omega \; + \; \frac{1}{\lambda^2} \, \| \, \nabla \cdot \sigma + f \, \|^2_\Omega$$

$$= \| \nabla x - \sigma \|_\Omega^2 - 2 \langle x, f + \nabla\cdot\sigma \rangle_\Omega + \lambda^2 \| x \|_\Omega^2 + \frac{1}{\lambda^2} \| \nabla\cdot\sigma + f \|_\Omega^2$$

$$= \| \nabla x - \sigma \|_\Omega^2 + \left\| \frac{\nabla\cdot\sigma + f}{\lambda} - \lambda x \right\|_\Omega^2 \tag{2.4}$$

for any vector field $\sigma \in H_{\rm div}(\Omega)$. Depending on our choice of this vector field different types of a posteriori estimates for the approximation error will result. We will revert to the question of how to construct a suitable field σ in a separate section below.

2.1.3 Error bounds for the Laplace problem

Before we apply the results summarised in section 1.2.7 to the Laplace problem with homogeneous Dirichlet boundary conditions, let us first fix our notation:

$$F(y) := \frac{1}{2} \int_\Omega \sum_{i=1}^n y_i(\xi)^2 \, d\xi \quad ; \quad y \in L^2(\Omega, \mathbb{R}^n)$$

$$G(x) := - \int_\Omega f(\xi)\, x(\xi) \, d\xi \quad ; \quad x \in L^2(\Omega) \quad ,$$

whereby $f \in L^2(\Omega)$ is an arbitrary function. Again, the symbol $\Lambda : X \longrightarrow Y$ denotes the gradient mapping $\nabla : H_0^1(\Omega) \longrightarrow L^2(\Omega, \mathbb{R}^n)$. The conjugate functional $G^* : L^2(\Omega) \longrightarrow \mathbb{R}$ reads:

$$G^*(x^*) = \begin{cases} 0 & ; \quad x^* + f = 0 \\ +\infty & ; \quad \text{else} \end{cases} .$$

As in the previous subsection we may employ the forcing function $\phi(t) = 1/4\, t^2$. Hence, for any vector field $\sigma \in H_{\rm div}(\Omega)$ the duality error estimate (1.8) has the following form:

$$\frac{1}{2} |\, x - x_0 \,|_{\Omega,1}^2 \leq \frac{1}{2} \| \nabla x \|_\Omega^2 + \frac{1}{2} \| \sigma \|_\Omega^2 - \langle x, f \rangle_\Omega + G^*(\nabla\cdot\sigma)$$

$$= \frac{1}{2} \| \nabla x - \sigma \|_\Omega^2 + G^*(\nabla\cdot\sigma) \quad . \tag{2.5}$$

Obviously, the above estimate is meaningless unless we can satisfy the admissibility condition

$$\sigma \in Z^* := \left\{ \tau \in H_{\rm div}(\Omega) \;\middle|\; \nabla\cdot\tau + f = 0 \right\} . \tag{2.6}$$

For all practical purposes it is therewith impossible to evaluate the duality majorant in such a way, that the right hand side of (2.5) stays finite. With a view to section 1.2.4 let us define:

$$\kappa\, H(y) := h(y) := F(y) \quad ; \quad y \in L^2(\Omega, \mathbb{R}^n)$$

with $\kappa > 0$ being some fixed parameter. A simple calculation demonstrates, that the function $h : L^2(\Omega, \mathbb{R}^n) \longrightarrow \mathbb{R}$ does meet the requirements set up by (1.10) and (1.11). In accordance with definition (1.19) we may therefore define the mapping $K : H_0^1(\Omega) \longrightarrow \mathbb{R}$ by:

$$K(s) := \inf_{y \in L^2(\Omega,\mathbb{R}^n)} \left\{ \frac{1}{\kappa} F(\nabla s - y) + F(y) \right\} = \frac{1}{2\,(\kappa+1)} \| \nabla s \|_\Omega^2 \quad ; \quad s \in H_0^1(\Omega) \ .$$

The conjugate mapping $K^* : H^{-1}(\Omega) \longrightarrow \mathbb{R}$ may be obtained in the following fashion:

$$K^*(s^*) = \sup_{s \in H_0^1(\Omega)} \left\{ \langle s, s^* \rangle_\Omega - \frac{\| \nabla s \|_\Omega^2}{2\,(\kappa+1)} \right\} = \sup_{t \geq 0} \left\{ \sup_{|s|_{\Omega,1} = 1} t \, \langle s, s^* \rangle_\Omega - \frac{t^2}{2\,(\kappa+1)} \right\}$$

$$= \sup_{t \geq 0} \left\{ t \, |\, s^* \,|_{\Omega,-1} - \frac{t^2}{2\,(\kappa+1)} \right\} = \frac{\kappa+1}{2} \, |\, s^* \,|_{\Omega,-1}^2 \quad ; \quad s^* \in H^{-1}(\Omega) \ .$$

We conclude, that in the present context the final energy error estimate (1.21) is equivalent to:

$$| x - x_0 |^2_{\Omega,1} \;\leq\; \frac{\kappa+1}{\kappa} \| \nabla x \,-\, \sigma \|^2_\Omega \;+\; (\kappa+1) \, | \nabla\cdot \sigma \,+\, f |^2_{\Omega,-1} \quad ; \quad \sigma \in L^2(\Omega, \mathbb{R}^n) \,. \tag{2.7}$$

Since the dual norm of the residual expression $\nabla\cdot\sigma + f$ is not readily available, the above a posteriori estimate cannot be evaluated directly. Still, we can find a computable upper bound to the quantity $|\nabla\cdot\sigma + f|_{\Omega,-1}$ with the help of those analytical techniques which are usually employed in the computation of residual based error estimators (see e. g. [5, 138]). Moreover, we may altogether avoid estimating the residual in the dual norm if we replace $K : H^1_0(\Omega) \longrightarrow \mathbb{R}$ by a suitable lower bound $\tilde{K} : L^2(\Omega) \longrightarrow \mathbb{R}$. Invoking Friedrich's inequality (1.4) we find:

$$\tilde{K}(s) \quad := \quad \frac{\lambda_0^2}{2\,(\kappa+1)} \, \| \, s \, \|^2_\Omega \quad \leq \quad K(s) \qquad ; \qquad s \in H^1_0(\Omega) \;\; .$$

Hence, if we bound K^* by the function $\tilde{K}^*$ we obtain the following *computable* error bound:

$$| x - x_0 |^2_{\Omega,1} \;\leq\; \frac{\kappa+1}{\kappa} \| \nabla x \,-\, \sigma \|^2_\Omega \;+\; \frac{\kappa+1}{\lambda_0^2} \, \| \nabla\cdot \sigma \,+\, f \|^2_\Omega \quad ; \quad \sigma \in H_{\mathrm{div}}(\Omega) \,. \tag{2.8}$$

2.1.4 On the Efficiency of the Error Estimates

Both the error bound (2.4) and the estimate (2.7) respectively (2.8) consist of two parts, which correspond to the two necessary and sufficient optimality conditions (1.12). In the case of the Helmholtz problem this correspondence is an immediate result of the method, by which it has been derived in section 1.2.2. That the formula (1.21) should provide an error estimate for the Laplace problem, which splits nicely into a residual in terms of the first optimality condition $\Lambda x \in \partial F^*(y^*)$ and another residual in terms of the second optimality condition $-\Lambda^* y^* \in \partial G(x)$ is not that obvious, however. We have not exploited the opportunity to choose a function H in (1.14) with a view to improving the error estimator to the fullest extent possible. Hence, the above expressions represent but one possible option in the design of computable a posteriori error estimates for the Laplace problem. Their structural simplicity may come at the cost of reduced effectiveness, since there is only the parameter $\kappa > 0$ in the definition of the function H to weigh the two parts of the error estimates.

Let us first look at the sharpest error bounds we might hope to obtain if the whole of the $L^2(\Omega, \mathbb{R}^n)$ respectively $H_{\mathrm{div}}(\Omega)$ were at our disposal. In the case of the Helmholtz problem we can recover the exact approximation error, if we choose $\sigma = \nabla x_0$. Our choice is feasible, as the data warrants $x_0 \in H^2(\Omega)$ thanks to the elliptic regularity of the problem. If we followed the same strategy in the case of the Laplace problem, the resulting error estimate need not necessarily be optimal, however. We note that we can represent any vector field $\sigma \in L^2(\Omega, \mathbb{R}^n)$ as a sum of two fields, the first of which is a potential flow while the second is solenoidal. That means, we can find a function $u \in H^1_0(\Omega)$ and some vector field $\sigma_{_D} \in L^2(\Omega)$ such that

$$\sigma \quad = \quad \nabla u \;\; + \;\; \sigma_{_D} \qquad \wedge \qquad \nabla\cdot\sigma_{_D} \;\; = \;\; 0$$

holds. Using the above representation, also known as the Helmholtz decomposition of the vector field σ, we can rewrite the error bound (2.8) in terms of the potential $u \in H^1_0(\Omega)$. We note:

$$\| \nabla x \,-\, \sigma \|^2_\Omega \quad = \quad \| \nabla x \,-\, \nabla u \|^2_\Omega \;\; + \;\; \| \, \sigma_{_D} \|^2_\Omega \;\; .$$

Hence, a stationary point of the right hand side of (2.8) is characterised by: $\sigma_{_D} = 0$. As the field σ is required to feature a square integrable divergence, we must assume that $\Delta u \in L^2(\Omega)$ holds. Due to the elliptic regularity of the Laplace problem our assumption implies: $u \in H^1_0(\Omega) \cap H^2(\Omega)$. Consequently, we may pose the following variational principle

$$\frac{\kappa+1}{\kappa} \, \langle \, \nabla u - \nabla x, \nabla v \, \rangle_\Omega \;\; + \;\; \frac{\kappa+1}{\lambda_0^2} \, \langle \, \Delta u + f, \Delta v \, \rangle_\Omega \;\; = \;\; 0 \qquad ; \qquad v \in H^1_0(\Omega) \cap H^2(\Omega)$$

to define the potential u and thence the stationary point $\sigma_0 := \nabla u$ of the error majorant (2.8). After performing a partial integration in the first of the above expressions we can exploit the fact

that the Laplace operator $\Delta : H^1_0(\Omega) \cap H^2(\Omega) \longrightarrow L^2(\Omega)$ is surjective. Thus we find, that the potential u satisfies the following differential equation in a strong sense:

$$-\Delta u \;+\; \frac{\lambda_0^2}{\kappa}\, u \;=\; f \;+\; \frac{\lambda_0^2}{\kappa}\, x \quad .$$

We conclude, that the right hand side of (2.8) may be seen as the dual formulation of the above Helmholtz problem, if we ignore certain scaling factors and additive constants. Inserting the vector field ∇u in (2.8) we obtain the following bound on the approximation error:

$$|\, x - x_0 \,|^2_{\Omega,1} \;\leq\; 2\, \tilde{M}_K(x, \sigma_0) \;=\; \frac{\kappa+1}{\kappa}\, \langle\, \nabla u - \nabla x\,,\, \nabla x_0 - \nabla x \,\rangle_\Omega \quad .$$

However, in the course of the very same computations we also find:

$$|\, x \,-\, u \,|^2_{\Omega,1} \;\leq\; \frac{2\,\kappa}{\kappa+1}\, \tilde{M}_K(x, \sigma_0) \;=\; \|\, \nabla x \,-\, \nabla u \,\|^2_\Omega \;+\; \frac{\lambda_0^2}{\kappa}\, \|\, x \,-\, u \,\|^2_\Omega \quad .$$

Therefore, we may combine these inequalities and thus derive an upper bound on the generalised hypercycle estimate $\tilde{M}_K$ in terms of the approximation error $|\, x - x_0 \,|_{\Omega,1}$:

$$2\, \tilde{M}_K(x, \sigma_0) \qquad \leq \qquad \frac{\kappa+1}{\kappa} \quad |\, x \,-\, x_0 \,|^2_{\Omega,1} \qquad .$$

We note, that we would have obtained exactly the same estimate for the efficiency of the error estimator (2.8), if we had chosen the vector field $\sigma = \nabla x_0$ on the right hand side of (2.8). However, the above analysis has demonstrated that we may view the quantity $\kappa^{-1} > 0$ as a perturbation parameter, whereby the linear functional $G : L^2(\Omega) \longrightarrow \mathbb{R}$ as it has been introduced in the very beginning of section 2.1.3 is replaced by:

$$G_\kappa(u) \quad := \quad G(u) \;+\; \frac{\lambda_0^2}{2\,\kappa} \int\limits_\Omega u(\xi)\, \big(\, u(\xi) - 2\, x(\xi) \,\big)\; d\,\xi \qquad ; \qquad u \in L^2(\Omega) \quad .$$

In the limit $\kappa \to +\infty$ the original variational formulation can be recovered. At the same time the efficiency index of the error bound (2.8) drops to 1, if the optimal choice $\sigma_0 \in H_{\mathrm{div}}(\Omega)$ is used for the dual vector field. In [16] a similar approach is developed to avoid any admissibility constraints to be imposed on the dual variable.

On the face of it, the estimate (2.7) seems to suffer from a loss of efficiency in the very same way its computable counterpart (2.8) does. However, there are some differences: The parameter $\kappa > 0$ acts as a penalty parameter for the admissibility constraint $\nabla \cdot \sigma + f = 0$ rather than a perturbation parameter, that affects the variational statement. Furthermore, the efficiency of the estimate is actually independent of our choice of that parameter. Since $-\Delta x_0 = f$ holds in a strong sense due to our assumptions on the data of the Laplace problem, we find indeed:

$$|\, f \,+\, \nabla\cdot \sigma \,|_{\Omega,-1} \quad = \quad \sup_{v \in H^1_0(\Omega)} \frac{\langle\, \nabla v, \nabla x_0 - \sigma \,\rangle_\Omega}{|\, v \,|_{\Omega,1}} \quad = \quad \|\, \nabla x_0 \,-\, \sigma \,\|_\Omega \quad .$$

Hence, the sharpest bound is obtained from (2.7) if the following variational principle is met:

$$\frac{\kappa+1}{\kappa}\, \langle\, \sigma - \nabla x, \tau \,\rangle_\Omega \;+\; (\kappa+1)\, \langle\, \sigma - \nabla x_0, \tau \,\rangle_\Omega \quad = \quad 0 \quad ; \quad \tau \in L^2(\Omega, \mathbb{R}^n) \quad .$$

From the above equation we conclude, that the right hand side of (2.7) attains the smallest value possible at the point $\sigma_0 \in L^2(\Omega, \mathbb{R}^n)$ which is defined by:

$$\sigma_0 \quad := \quad \frac{\kappa}{1+\kappa} \left\{\, \nabla x_0 \;+\; \frac{1}{\kappa}\, \nabla x \,\right\} \quad .$$

Evaluating the error majorant (2.7) at this point we find after a few algebraic manipulations:

$$\left\{\, \frac{\kappa+1}{\kappa}\, \|\, \nabla x \,-\, \sigma \,\|^2_\Omega \;+\; (\kappa+1)\; |\, \nabla\cdot \sigma \,+\, f \,|^2_{\Omega,-1} \,\right\} \Big|_{\sigma=\sigma_0} \quad = \quad |\, x \,-\, x_0 \,|^2_{\Omega,1} \quad .$$

As the error majorant (1.25) provides an upper bound for the alternative estimate (1.26), we conclude that the latter majorant can also be used to recover the approximation error $|\, x - x_0 \,|_{\Omega,1}$ exactly. In fact, on closer examination of the more accurate error estimate we perceive in the present instance, that the quantity $M_K(x,\sigma)$ is independent of the dual variable $\sigma \in L^2(\Omega,\mathbb{R}^n)$. Against the proof of our claim let us introduce the following space:

$$H_0(\Omega) \quad := \quad \left\{ \; \rho \in H_{\rm div}(\Omega) \;\middle|\; \nabla\cdot \rho \;=\; 0 \; \right\}$$

which contains all square-integrable, solenoidal vector fields. Subsequently, we may introduce the self-adjoint projection operator $Q : L^2(\Omega,\mathbb{R}^n) \longrightarrow H_0(\Omega)^\perp$ by requiring:

$$\langle\, Q\,\tau \,,\, \nabla v \,\rangle_\Omega \quad = \quad \langle\, \tau \,,\, \nabla v \,\rangle_\Omega \qquad ; \qquad v \in H_0^1(\Omega)$$

with $\tau \in L^2(\Omega,\mathbb{R}^n)$ being an arbitrary vector field. We note, that we can find for any element $\tau \in H_0(\Omega)^\perp$ an unique potential $w_\tau \in H_0^1(\Omega)$, such that $\tau = \nabla w_\tau$ holds. We may define a second projection $P : L^2(\Omega,\mathbb{R}^n) \longrightarrow H_0(\Omega)$ by requiring: $P\,\tau := \tau - Q\,\tau$ for any $\tau \in L^2(\Omega,\mathbb{R}^n)$. Let us now consider some distribution $x^* \in H^{-1}(\Omega)$. The variational problem

$$\langle\, \nabla w_x, \nabla v \,\rangle_\Omega \quad = \quad x^*(v) \qquad ; \qquad v \in H_0^1(\Omega)$$

is known to possess an unique solution $w_x \in H_0^1(\Omega)$. Hence, there is a vector field $\xi \in H_0(\Omega)^\perp$, namely ∇w_x, such that $-\nabla\cdot\xi = x^*$ holds in a distributional sense. We state:

$$K_\tau^*(x^*) \;=\; \sup_{x \in H_0^1(\Omega)} \left\{ \,\langle\, x, x^* \,\rangle_\Omega - \frac{1}{2}\,\|\, \tau + \nabla x \,\|_\Omega^2 \,\right\} \;=\; \sup_{x \in H_0^1(\Omega)} \left\{ \,\langle\, \nabla x, \xi \,\rangle_\Omega - \frac{1}{2}\,\|\, \tau + \nabla x \,\|_\Omega^2 \,\right\}$$

$$= \sup_{\rho \in L^2(\Omega,\mathbb{R}^n)} \left\{ \,\langle\, \rho, \xi \,\rangle_\Omega \;-\; \frac{1}{2}\,\|\, \tau + Q\,\rho \,\|_\Omega^2 \,\right\} \qquad ; \qquad \tau \in L^2(\Omega,\mathbb{R}^n) \;\; .$$

The last expression on the right hand side of the above equation features a stationary point $\rho_0 \in L^2(\Omega,\mathbb{R}^n)$, which the functional value $K_\tau^*(x^*)$ is attained at. By a simple computation we find that this point is defined by the following necessary optimality condition:

$$\xi \quad = \quad Q^*\,(\,\tau \,+\, Q\,\rho_0\,) \quad = \quad Q\,\tau \,+\, Q\,\rho_0 \;\; .$$

To evaluate $K_\tau^*(x^*)$ we exploit, that $Q\,\xi = \xi$ holds. Thus we may infer from the above condition:

$$K_\tau^*(x^*) \;=\; \langle\, Q\,\rho_0 \,,\, \xi \,\rangle_\Omega - \frac{1}{2}\,\|\, P\,\tau \,+\, \xi \,\|_\Omega^2 \;=\; \frac{1}{2}\,\|\, \xi \,\|_\Omega^2 \;-\; \langle\, \tau, \xi \,\rangle_\Omega \;-\; \frac{1}{2}\,\|\, P\,\tau \,\|_\Omega^2 \;\; .$$

Combining this result with the definition (1.27) of the improved error bound $M_K(x,\sigma)$ we note:

$$M_K(x,\sigma) \;=\; \frac{1}{2}\,\|\, \sigma \,-\, \nabla x \,\|_\Omega^2 \;+\; \frac{1}{2}\,\|\, \xi \,\|_\Omega^2 \;-\; \langle\, \sigma - \nabla x, \xi \,\rangle_\Omega \;-\; \frac{1}{2}\,\|\, P\,(\,\sigma \,-\, \nabla x\,) \,\|_\Omega^2$$

$$= \; \frac{1}{2}\,\|\, \sigma \,-\, \nabla x \,-\, \xi \,\|_\Omega^2 \;-\; \frac{1}{2}\,\|\, P\,\sigma \,\|_\Omega^2 \;=\; \frac{1}{2}\,\|\, Q\,\sigma \,-\, \nabla x \,-\, \xi \,\|_\Omega^2 \;\; ,$$

whereby the vector field $\xi \in H_0(\Omega)^\perp$ is defined by the requirement $\nabla\cdot\xi = \nabla\cdot\sigma + f$ to be met in a distributional sense. In order to eliminate the potential ξ from the right hand side of the above identity let us rewrite the norm of the vector field:

$$\|\, Q\sigma - \nabla x - \xi \,\|_\Omega \;=\; \sup_{\eta \in L^2(\Omega,\mathbb{R}^n)} \frac{\langle\, Q\,\eta \,,\, Q\,\sigma - \nabla x - \xi \,\rangle_\Omega}{\sqrt{\|\, Q\,\eta \,\|_\Omega^2 \,+\, \|\, P\,\eta \,\|_\Omega^2}} \;=\; \sup_{v \in H_0^1(\Omega)} \frac{\langle\, \nabla v \,,\, Q\,\sigma - \nabla x - \xi \,\rangle_\Omega}{|\, v \,|_{\Omega,1}}$$

$$= \; |\, \nabla\cdot\sigma \,+\, f \,+\, \Delta x \,-\, \nabla\cdot Q\,\sigma \,|_{\Omega,-1} \;=\; |\, f \,+\, \Delta x \,|_{\Omega,-1} \;\; .$$

Herewith, the proof of our claim has been finished.

2.1.5 The relationship with the Helmholtz problem

The computable error majorant (2.8) is closely related to the hypercycle estimate (2.4). If we replace the forcing function $f \in L^2(\Omega)$, which appears in the statement of the Helmholtz problem, with the shifted function $f + \lambda^2 x$, the resulting error bound has the very form of the estimate (2.8). We note, that the Laplace and the Helmholtz problem share the same numerical solution, albeit their analytical solutions are different. Hence, we must combine the hypercycle estimate (2.4) with an a priori estimate for the Helmholtz equation, if we want to derive an a posteriori error estimator for the Laplace problem. Let $\tilde{x}_0 \in H^1_0(\Omega)$ denote the analytical solution of the Helmholtz problem. Since $x_0 \in H^1_0(\Omega)$ solves the equation

$$-\Delta x \quad + \quad \lambda^2 x \quad = \quad f \; + \; \lambda^2 x_0$$

we may state the following result on the distance between x_0 and $\tilde{x}_0$:

$$| x_0 - \tilde{x}_0 |^2_{\Omega,1} \quad \leq \quad | x_0 - \tilde{x}_0 |^2_{\Omega,1} \; + \; \lambda^2 \, \| x_0 - \tilde{x}_0 \|^2_\Omega \quad \leq \quad \lambda^2 \, | x - x_0 |_{\Omega,-1} \, | x_0 - \tilde{x}_0 |_{\Omega,1} \quad .$$

Therefore, the a posteriori estimate for the approximation error reads:

$$| x - x_0 |_{\Omega,1} \quad \leq \quad \lambda^2 \, | x - x_0 |_{\Omega,-1} \quad + \quad \sqrt{\| \nabla x - \sigma \|^2_\Omega \; + \; \lambda^{-2} \, \| \nabla \cdot \sigma + f \|^2_\Omega} \tag{2.9}$$

with $\sigma \in H_{\mathrm{div}}(\Omega)$ being an arbitrary vector field. Thanks to Friedrich's inequality we can bound the dual norm of the approximation error in terms of its energy norm:

$$| x - x_0 |_{\Omega,-1} \quad = \quad \sup_{u \in H^1_0(\Omega)} \frac{(x - x_0, u)_\Omega}{| u |_{\Omega,1}} \quad \leq \quad \lambda_0^{-1} \, \| x - x_0 \|_\Omega \quad \leq \quad \lambda_0^{-2} \, | x - x_0 |_{\Omega,1} \quad .$$

Hence, we may hope to absorb the dual norm of the approximation error into the left hand side of the estimate (2.9), if the perturbation parameter λ is but small enough. Accordingly, let us introduce a penalty parameter $\kappa > 0$ by requiring:

$$\kappa \quad := \quad \frac{\lambda_0^2}{\lambda^2} \quad - \quad 1 \quad .$$

We note, that the above definition does not impose any undue restrictions on our choice of the perturbation parameter λ, since we must ensure $\lambda > \lambda_0$ in any case to retain a positive factor in front of the energy norm. By a simple computation we eventually find:

$$| x - x_0 |^2_{\Omega,1} \quad \leq \quad \frac{(\kappa + 1)^2}{\kappa^2} \; \| \nabla x - \sigma \|^2_\Omega \quad + \quad \frac{(\kappa + 1)^3}{\kappa^2 \, \lambda_0^2} \; \| \nabla \cdot \sigma + f \|^2_\Omega \quad . \tag{2.10}$$

We infer from (2.10) that the estimate for the approximation error we can obtain from a perturbed variational formulation is less accurate than the generalised hypercycle estimate (2.8) we have derived by a penalisation of the admissibility constraints as outlined in section 1.2.4. Using the variational statement of the Helmholtz problem instead of the Dirichlet integral is a device, which has already been exploited in [16]. The discussion found in [5] is similar to our analysis and can be subsumed under our approach when we suppose, $x \in H^1_0(\Omega)$ designates the numerical solution of a Galerkin scheme. In this special case, the interpolation properties of the finite element ansatz warrant the following bound on the dual norm:

$$| x - x_0 |_{\Omega,-1} \quad \leq \quad C \, h \, \lambda_0^{-1} \; | x - x_0 |_{\Omega,1} \quad .$$

Hereby, $h > 0$ denotes the diameter of the largest finite element and C designates a constant which depends on certain geometrical properties of the mesh only. The meaning of interpolation estimates such as the above will be explained more fully in section 3.1.2.

2.1.6 Summary Statement of our Results

In the previous sections we have considered four different error bounds: One for the Helmholtz and three for the Laplace problem. We have seen, that the Helmholtz problem allows for an

computable error majorant, that can be obtained immediately from the data of the variational statement, since there are no admissibility constraints involved in the statement of the dual formulation. The three hypercycle estimates we have derived for the Laplace problem suffer from the necessity of accounting for the admissibility constraint $\nabla \cdot \sigma + f = 0$ which the dual variable $\sigma \in L^2(\Omega, \mathbb{R}^n)$ must meet in a weak sense. Applying the theoretical results we have developed throughout section 1.2 we have relaxed the admissibility constraint. Despite that, the error majorants M_K and $\hat{M}_K$ have proved sharp in the following sense:

$$| x - x_0 |^2_{\Omega,1} \quad = \quad 2\, M_K(x, \sigma) \quad = \quad \inf_{\tau \in L^2(\Omega, \mathbb{R}^n)} 2\, \hat{M}_K(x, \tau) \qquad ; \qquad \sigma \in L^2(\Omega, \mathbb{R}^n) \ .$$

However, the above result is but of little practical value, since there is no closed analytical expression for the error majorant M_K. The second majorant $\hat{M}_K$ involves some residual in the second duality relation, which is measured in the norm of the appropriate dual space. Hence, we need to bound the majorant with the help of those analytical techniques, usually employed in the treatment of conventional residual based error estimators (see e. g. [5, 138]).

We conclude, that we can compute bounds on the majorant $\hat{M}_K$ only under those assumptions we must impose when we want to evaluate a conventional a posteriori error estimator based on element residuals: namely, the best approximation property of the numerical solution and the absence of any cubature errors. The only a posteriori bound on the approximation error which is readily computable is (2.8). We have seen, that this bound can in turn be limited by the approximation error. At least in principle, the efficiency index of the error majorant reads $1+\kappa^{-1}$, whereby $\kappa > 0$ may be interpreted as a perturbation parameter for a Helmholtz problem, whose solution defines the optimal dual parameter σ. In the limit $\kappa \to \infty$ the original Dirichlet problem emerges. At the same time, the admissibility constraint $\nabla \cdot \sigma + f = 0$ is reintroduced due to the penalisation and the conventional hypercycle estimate is recovered.

Unfortunately, the exact solution of the auxiliary problem is not available, whence the optimal parameter must be approximated itself by a vector field from some finite dimensional trial space. The analysis of practical choices for the parameter σ is beyond the scope of this paragraph and shall be deferred to section 3.3.1. The possibility of using complementary energy principles for the Helmholtz problem in order to obtain error bounds for the Dirichlet problem has been discussed e. g. in [5, 16]. The resulting error bounds have been found inferior to the estimate (2.8), unless we assume, that the numerical solution x has the best approximation property.

2.2 The Obstacle Problem

Though in a sense the obstacle problem may be as generic an example for a constrained variational formulation as the Laplace problem is a generic example for an unconstrained formulation, the obstacle problem or variants thereof have nevertheless quite a number of important applications from such diverse realms as engineering sciences or financial mathematics. To name but a few of these applications we may mention contact problems in elastomechanics, saturation problems in porous media, the treatment of cavitation phenomena and the pricing of financial derivatives.

In the first section we will briefly discuss two mathematical concepts, which are requisite in order to properly pose the obstacle problem. We define capacities of sets and order relations between functions from the spaces $W^{1,p}(\Omega)$. In the following section the variational formulation is presented and the appertaining notation fixed. In section 2.2.3 we digress slightly to discuss the meaning of the subdifferential ∂G and to introduce further notation. Our abstract framework is brought to bear on the obstacle problem in section 2.2.5. An alternative approach which has been detailed in a technical report [41, 42] is presented though briefly in the next section. Some remarks on the efficiency of the emergent a posteriori estimates for the approximation error are collected in section 2.2.6: they are intended to prepare the reader against our analysis of these estimates in the section 2.2.7. Our findings are reviewed in section 2.2.8.

2.2.1 Capacity and order relations on Sobolev spaces

It is a well known fact, that the point values of Lebesgue functions are not uniquely defined. That means, two Lebesgue function $f, f' \in L^p(\Omega)$ are deemed identical, if they differ but on a set of Lebesgue measure 0. Due to the ambiguity, which is inherent in their definition, the concept of

some function $f \in W^{1,p}(\Omega)$ being larger than some other function $f' \in W^{1,p}(\Omega)$ on a set $E \subset \Omega$ needs to be elaborated before we can properly state the obstacle problem. We will assume, that set $E \subset \Omega$ is closed. However, we do not even require the set E to have a positive measure. (If the measure of E vanishes, the "obstacle" is usually referred to as *thin*.) To understand why an obstacle function $\psi \in C^1(\Omega)$ can affect the solution of the variational problem

$$x_0 \in V_\psi \quad : \quad \langle \nabla x_0, \nabla x - \nabla x_0 \rangle_\Omega \quad \geq \quad \langle f, x - x_0 \rangle_\Omega \quad ; \quad x \in V_\psi$$

when the cone $V_\psi \subset H^1_0(\Omega)$ is defined in terms to be presently resolved by

$$V_\psi \quad := \quad \left\{ \, v \in H^1_0(\Omega) \;\middle|\; v \geq \psi \text{ on } E \text{ in the sense of } H^1(\Omega) \, \right\} \tag{2.11}$$

even if E is a set of measure zero, we need to introduce the concept of *capacity*. With the help of this very concept we will be able to formalise our notion of an order relation in the Banach space $W^{1,p}(\Omega)$. For any set $A \subset \Omega$ let us specify the following set of functions:

$$\mathfrak{V}_p(A) \quad := \quad \left\{ \, v \in W^{1,p}(\Omega) \;\middle|\; v\,|_U \geq 1 \text{ on some neighbourhood } U \supset A \, \right\} \; .$$

We note, that in the above definition the requirement $v \geq \psi$ is to be understood in the usual sense, that is, $v \geq \psi$ is supposed to hold almost everywhere in the neighbourhood U of the set A. We can proceed and define the p-capacity of A in accordance with [57] by:

$$\mathrm{cap}_p(A) \quad := \quad \inf_{v \in \mathfrak{V}_p(A)} \int\limits_\Omega \sum_{i=1}^n \left| \frac{\partial v}{\partial \xi_i}(\xi) \right|^p \, d\xi \quad .$$

We will call a function $f : \Omega \longrightarrow \mathbb{R}$ *quasi-continuous* on Ω, if we can find for any $\epsilon > 0$ a subset $E_\epsilon \subset \Omega$ with the property $\mathrm{cap}_p(E_\epsilon) < \epsilon$, such that f is continuous on $\Omega \setminus E_\epsilon$.

It is a well known fact, that each function $f \in W^{1,p}(\Omega)$ has a quasi-continuous representative (see e. g. [64, 108]). Hence, the point values of $f \in W^{1,p}(\Omega)$ are uniquely defined up to some set of capacity zero. We infer, that it is not sufficient to consider a function $f \in W^{1,p}(\Omega)$ positive on the set E, if $f|_E \geq 0$ holds almost everywhere. In fact, we have to insist that f be positive everywhere on E except for a set of capacity zero. Accordingly, we shall call $f \in W^{1,p}(\Omega)$ greater or equal $f' \in W^{1,p}(\Omega)$ on $E \subset \Omega$ in the sense of $W^{1,p}(\Omega)$, if $f - f' \geq 0$ holds everywhere on E except for some subset of capacity zero.

An equivalent definition of the cone V_ψ introduced in (2.11) can be given in terms of sequences of continuous functions. Indeed, we may call a function $f \in H^1(\Omega)$ greater or equal than $\psi \in H^1(\Omega)$ on the set $E \subset \overline{\Omega}$ in the sense of $H^1(\Omega)$, if there is sequence $\{u_n\}_n \subset C^1(\overline{\Omega})$ such that:

$$u_n\big|_E \;\geq\; 0 \quad ; \quad n \in \mathbb{N} \qquad \wedge \qquad \lim_{n \to \infty} \| \, f - \psi - u_n \, \|_{\Omega,1} \;=\; 0 \quad .$$

Let us remark, that the order relation we have introduced for the Sobolev spaces $W^{1,p}(\Omega)$ ensures, the convex cone V_ψ as defined by (2.11) is closed, if E is a closed subset of the domain Ω.

2.2.2 Statement of the variational formulation

To formalise the obstacle problem we can adopt practically all of the notation we have introduced in the sections 2.1.1 and 2.1.2. Let us aim, however, at a slightly more general approach to constrained variational problems. We introduce a matrix valued mapping $A \in W^{1,\infty}(\Omega, \mathbb{R}^{n \times n}_{\mathrm{sym}})$ which satisfies the following ellipticity condition:

$$A_0 > 0 \quad : \quad \sum_{i=1}^n \sum_{j=1}^n A_{ij}(\xi) \, v_i \, v_j \;\geq\; A_0 \, \|v\|^2 \quad ; \quad v \in \mathbb{R}^n \, , \; \xi \in \Omega \; .$$

Furthermore, we assume that the functions $\psi \in H^2(\Omega)$ constitutes the "obstacle" and defines the cone $V_\psi \subset H^1_0(\Omega)$ of admissible functions in accordance with (2.11). Until further notice, let us suppose, that set E and the domain Ω coincide. The boundary conditions are determined by our choice of the function space $X = H^1_0(\Omega)$. Hence, we have to impose some restrictions on ψ in order to prevent the contingency $V_\psi = \emptyset$. We require: $\psi|_{\partial\Omega} < 0$ everywhere on the boundary

of the domain Ω. The point values of ψ are well defined, since $H^2(\Omega) \hookrightarrow C(\overline{\Omega})$ holds by the imbedding theorem of Sobolev (see e. g. [2]). Setting $Y = L^2(\Omega, \mathbb{R}^n)$ let us define:

$$F(y) \quad := \quad \frac{1}{2} \int\limits_{\Omega} \Big\{ \sum_{i=1}^{n} \sum_{j=1}^{n} A_{ij}(\xi)\, y_i(\xi)\, y_j(\xi) \Big\}\, d\,\xi \qquad ; \qquad y \in L^2(\Omega, \mathbb{R}^n) \ .$$

Clearly, the functional F is uniformly convex. A suitable forcing function $\phi : \mathbb{R}_0^+ \longrightarrow \mathbb{R}_0^+$, which may be employed in (1.6), is given by: $\phi(t) := A_0/4t^2$. To represent the condition $u_0 \in V_\psi$ within the framework we have outlined in paragraph 1.2.1 let us complement the functional G with the indicator function of the cone V_ψ:

$$G(x) \quad := \quad \chi_{V_\psi}(x) \ - \int\limits_{\Omega} f(\xi)\, x(\xi)\, d\,\xi \qquad ; \qquad x \in H_0^1(\Omega) \quad .$$

We shall assume, that $f \in L^2(\Omega)$ holds. As in the previous section the operator $\Lambda : X \longrightarrow Y$ will designate the gradient mapping. Thus, we can identify the variational inequality

$$x_0 \in V_\psi \qquad : \qquad \langle\, A\, \nabla x_0, \nabla x - \nabla x_0 \,\rangle_\Omega \quad \geq \quad \langle\, f, x - x_0 \,\rangle_\Omega \qquad ; \qquad x \in V_\psi$$

as the necessary and sufficient optimality condition for the following minimisation task:

$$F(\Lambda x) \ + \ G(x) \quad \underset{x \in V_\psi}{\longrightarrow} \quad \inf \quad . \tag{2.12}$$

We remark, that the indicator function of the cone V_ψ is convex and lower-semicontinuous, as the set $V_\psi \subset H_0^1(\Omega)$ is convex and closed. Therefore, the theory we have developed throughout the first chapter is applicable indeed.

2.2.3 A Description of the Subdifferential $\partial G(x)$

Let $x \in V_\psi$ denote an arbitrary function. In the following we want to discuss which conditions a distribution $x^* \in H^{-1}(\Omega)$ must meet in order to be contained in the set $\partial G(x)$. With a view to paragraph 1.1.5 we may state right away, that any $x^* \in \partial G(x)$ must satisfy the inequality

$$G(x') \quad \geq \quad \int\limits_{\Omega} \Big\{\, x^*(\xi)\, \{\, x'(\xi) - x(\xi)\,\} \ - \ f(\xi)\, x(\xi) \,\Big\}\, d\,\xi \qquad ; \qquad x' \in H_0^1(\Omega) \ .$$

Since the left hand side is only finite, if $x' \in V_\psi$ holds, we may replace the above inequality by:

$$0 \quad \geq \quad \int\limits_{\Omega} \big(\, x'(\xi) - x(\xi) \,\big)\, \big(\, f(\xi) \ + \ x^*(\xi) \,\big)\, d\,\xi \qquad ; \qquad x' \in V_\psi \ . \tag{2.13}$$

If some function $x' \in H_0^1(\Omega)$ is greater or equal than $x \in V_\psi$ in the sense of $H^1(\Omega)$, it is contained in the cone V_ψ by construction. Therefore, we may infer from (2.13) that $-f - x^*$ constitutes a positive measure. Let us assume, that we can find a point $\xi_0 \in \Omega$ and a number $r > 0$, such that there is a function $u_0 \in C^1(\Omega)$, whose support is contained in the ball $B_{2r}(\xi_0)$ and which satisfies $u_0 > 0$ on $B_r(\xi_0)$ along with the requirement $x \geq \psi + u_0$ in the sense of $H^1(\Omega)$. We shall say, that $x > \psi$ holds at the point ξ_0 in the sense of $H^1(\Omega)$.

Let $\vartheta \in C_0^1(B_r(\xi_0))$ denote an arbitrary function. By definition we can find a scaling factor $\vartheta_0 > 0$ such that $|\, \vartheta_0\, \vartheta \,| \leq u_0$ holds on $B_r(\xi_0)$. Hence, both $x + \vartheta_0\, \vartheta$ and $x - \vartheta_0\, \vartheta$ belong to the cone V_ψ. With a view to (2.13) we conclude using a density argument:

$$0 \quad = \quad \int\limits_{B_r(\xi_0)} \vartheta(\xi)\, \big(\, f(\xi) \ + \ x^*(\xi) \,\big)\ d\,\xi \qquad ; \qquad \vartheta \in H_0^1(B_r(\xi_0)) \ .$$

Thanks the above result we may reason, that the support of the measure $-f - x^*$ is contained within the set of those points, where the obstacle ψ and the function x have come into "contact". To formalise the notion of the *coincidence set* let us define:

$$\Omega_x \quad := \quad \Omega \ \setminus \ \Big\{\, \xi \in \Omega \,\Big|\, x > \psi \text{ at } \xi \text{ in the sense of } H^1(\Omega) \,\Big\} \quad .$$

In the case $\Omega \setminus E \neq \emptyset$ we may apply a similar reasoning to any point $\xi \in \Omega$ that has a positive distance to the set E. Thus we find, that any distribution $x^* \in \partial G(x)$ can be represented with the help of some positive measure μ, whose support is confined to the set $\Omega_x \cap E$:

$$\langle\, s, x^* \,\rangle_\Omega \quad = \quad -\,\langle\, s, f \,\rangle_\Omega \;-\; \int\limits_{\Omega_x \cap E} s \;\, d\,\mu \qquad ; \qquad s \in H_0^1(\Omega) \;\; . \tag{2.14}$$

Conversely, let us assume, that μ is a positive measure, whose support is contained in $\Omega_x \cap E$. Obviously, the right hand side of (2.14) defines a distribution $x^* \in H^{-1}(\Omega)$. We are going to demonstrate, that $x^* \in \partial G(x)$ holds. For any function $x' \notin V_\psi$ there is nothing to prove. Hence, let us suppose $x' \in V_\psi$. We note:

$$G(x) \;+\; \langle\, x' - x, x^* \,\rangle_X \quad = \quad G(x') \;-\; \int\limits_{\Omega_x \cap E} \big(\, x' - \psi \,\big) \;\, d\,\mu \quad \leq \quad G(x') \;\; .$$

From the above inequality we infer, that x^* is in fact a subgradient of G at the point $x \in V_\psi$. Henceforth, we shall say the distribution $z^* \in H^{-1}(\Omega)$ belongs to the set $\mathfrak{M}_x$ if there is a positive measure ζ, such that the action of z^* can be described by:

$$\langle\, s, z^* \,\rangle_\Omega \quad := \quad \int\limits_{E \,\cap\, \Omega_x} s \;\; d\,\zeta \qquad ; \qquad s \in H_0^1(\Omega) \;\; .$$

2.2.4 Error estimates for the Energy Norm

Thanks to the ellipticity condition imposed on the function $A \in W^{1,\infty}(\Omega, \mathbb{R}^{n\times n}_{\rm sym})$ we can warrant the existence of another function $A^{-1} \in W^{1,\infty}(\Omega, \mathbb{R}^{n\times n}_{\rm sym})$ such that $A\,A^{-1} \equiv I \in \mathbb{R}^{n\times n}$ holds. We note that this "inverse" function is also coercive, as we find at any point $\xi \in \Omega$:

$$\|\, A^{-1}(\xi)\, v \,\|_2 \quad = \quad \sup_{u \in \mathbb{R}^n} \frac{u^T A^{-1}(\xi)\, v}{\|u\|_2} \quad \geq \quad \sup_{u \in \mathbb{R}^n} \frac{u^T\, v}{\|A(\xi)\|_2 \|u\|_2} \quad \geq \quad \frac{\|v\|_2}{n\,\|A(\xi)\|_\infty} \qquad ; \qquad v \in \mathbb{R}^n \;\; .$$

With the help of the matrix valued mapping A^{-1} we can exhibit the Fenchel conjugate of the functional F, as it has been defined in subsection 2.2.2:

$$F^*(y^*) \quad := \quad \frac{1}{2} \int\limits_\Omega \Big\{ \sum_{i=1}^n \sum_{j=1}^n A^{-1}_{ij}(\xi)\, y_i^*(\xi)\, y_j^*(\xi) \Big\} \;\, d\,\xi \qquad ; \qquad y^* \in L^2(\Omega, \mathbb{R}^n) \;\; .$$

With a view to paragraph 2.1.3 let us introduce a parameter $\kappa > 0$ and let us define the auxiliary functions $h : L^2(\Omega, \mathbb{R}^n) \longrightarrow \mathbb{R}$ and $H : L^2(\Omega, \mathbb{R}^n) \longrightarrow \mathbb{R}$ by:

$$\kappa\, H(y) \quad := \quad h(y) \quad := \quad F(y) \qquad ; \qquad y \in L^2(\Omega, \mathbb{R}^n) \;\; .$$

We can verify easily, that the function h meets the requirements imposed by (1.10) and (1.11). With a view to definition (1.19) we proceed and introduce the mapping $K : H_0^1(\Omega) \longrightarrow \mathbb{R}$ by:

$$K(s) \quad := \quad \inf_{y \in L^2(\Omega, \mathbb{R}^n)} \Big\{ \frac{1}{\kappa} F(\nabla s - y) \;+\; F(y) \Big\} \quad = \quad \frac{\langle\, \nabla s, A\, \nabla s \,\rangle_\Omega}{2\,(\kappa + 1)} \qquad ; \qquad s \in H_0^1(\Omega) \;\; .$$

A bound on its Fenchel conjugate $K^* : H^{-1}(\Omega) \longrightarrow \mathbb{R}$ may be obtained in the following manner:

$$K^*(s^*) \quad = \quad \sup_{t \,\geq\, 0} \sup_{|s|_{\Omega,1} = 1} \Big\{ t\, \langle\, s, s^* \,\rangle_\Omega \;-\; t^2\, \frac{\langle\, \nabla s, A\, \nabla s \,\rangle_\Omega}{2\,(\kappa + 1)} \Big\} \quad = \quad \sup_{|s|_{\Omega,1} = 1} \Big\{ \frac{(\kappa + 1)\, \langle\, s, s^* \,\rangle_\Omega^2}{2\, \langle\, \nabla s, A\, \nabla s \,\rangle_\Omega} \Big\}$$

$$\leq \quad \sup_{|s|_{\Omega,1} = 1} \Big\{ \frac{\kappa + 1}{2\, A_0}\, \frac{\langle\, s, s^* \,\rangle_\Omega^2}{\|\nabla s\|_\Omega^2} \Big\} \quad = \quad \frac{\kappa + 1}{2\, A_0}\, |\, s^* |^2_{\Omega,-1} \qquad ; \qquad s^* \in H^{-1}(\Omega) \;\; .$$

Combining our findings from paragraph 2.2.3 with the above result we can derive from (1.21) an estimate of the energy error we encounter in approximating the solution $x_0 \in V_\psi$ of the variational problem (2.12) with some function $x \in V_\psi$:

$$| x - x_0 |^2_{\Omega,1} \quad \leq \quad \frac{2}{A_0} \, \frac{1+\kappa}{\kappa} \, F(\nabla x - A^{-1}\sigma) \quad + \quad \frac{\kappa+1}{A_0^2} \, \inf_{\mu \in \mathfrak{M}_x} | \mu + f + \nabla\cdot \sigma |^2_{\Omega,-1} \qquad (2.15)$$

which is reliable for any choice of the vector field $\sigma \in L^2(\Omega, \mathbb{R}^n)$. To obtain an error bound that is more amenable to numerical computation we must find a suitable lower bound $\tilde{K} : L^2(\Omega) \longrightarrow \mathbb{R}$ for the functional K. Exploiting the ellipticity condition stated in paragraph 2.2.2 we find by invoking Friedrich's inequality (1.4):

$$\tilde{K}(s) \quad := \quad \frac{A_0 \, \lambda_0^2}{2\,(\kappa+1)} \, \| \, s \, \|^2_\Omega \quad \leq \quad K(s) \qquad ; \qquad s \in H^1_0(\Omega) \ .$$

In analogy to (2.8) we arrive at the following computable error estimate for $\sigma \in H_{\mathrm{div}}(\Omega)$:

$$\begin{aligned} | x - x_0 |^2_{\Omega,1} \quad &\leq \quad \frac{2}{A_0} \, \frac{1+\kappa}{\kappa} \, F(\nabla x - A^{-1}\sigma) \, + \, \frac{\kappa+1}{A_0^2 \, \lambda_0^2} \, \inf_{\mu \in L^2(\Omega) \cap \mathfrak{M}_x} \| \, \mu + f + \nabla\cdot \sigma \, \|^2_\Omega \qquad (2.16) \\ &\leq \quad \frac{2}{A_0} \, \frac{1+\kappa}{\kappa} \, F(\nabla x - A^{-1}\sigma) \, + \, \frac{\kappa+1}{A_0^2 \, \lambda_0^2} \big\{ \| \, f + \nabla\cdot \sigma \, \|^2_\Omega - \| \, 0 \, \wedge \, (\nabla\cdot \sigma + f) \, \|^2_{\Omega_x \cap E} \big\} \, . \end{aligned}$$

Hereby, the symbol $u \wedge v$ denotes the common greatest lower bound of two functions $u, v \in L^2(\Omega)$.

2.2.5 An alternative approach to the obstacle problem

In a joint work with S. Repin [42] a somewhat different method has been developed to obtain computable a posteriori error estimates for the obstacle problem. The mathematical technology necessary to compute such estimates has been discussed in great detail in a technical note [41]. Hence, we will give only a very brief survey of our results. The starting point of our investigation is the Lagrangian $\mathcal{L} : H^1_0(\Omega) \times L^2(\Omega, \mathbb{R}^n) \times \mathfrak{M}_\psi \longrightarrow \mathbb{R}$ defined by:

$$\mathcal{L}(v, \tau, \mu^*) \quad := \quad \int_\Omega \Big\{ \, \tau \nabla v \, - \, \frac{1}{2} \, \tau^T A^{-1} \tau \, - \, f \, v \, \Big\} \, d\xi \quad - \quad \langle \, v - \psi, \, \mu^* \, \rangle_E \ .$$

While the energy functional $J : H^1_0(\Omega) \longrightarrow \mathbb{R}$ associated with the primal formulation (2.12) can be recovered by computing the supremum of $\mathcal{L}$ with respect to the second and third argument

$$J(v) \quad = \quad \sup_{\tau \in L^2(\Omega,\mathbb{R})} \, \sup_{\mu^* \in \mathfrak{M}_\psi} \, \mathcal{L}(v, \tau, \mu^*) \quad = \quad \begin{cases} \frac{1}{2} \, \langle \, \nabla v, A \nabla v \, \rangle_\Omega \, - \, \langle \, v, f \, \rangle_\Omega & ; \quad v \in V_\psi \\ + \, \infty & ; \quad \text{else} \end{cases} ,$$

the dual functional $J^* : L^2(\Omega, \mathbb{R}^n) \times \mathfrak{M}_\psi \longrightarrow \mathbb{R}$ is obtained by finding the minimum of the Lagrangian $\mathcal{L}$ with respect to its first argument:

$$J^*(\tau, \mu^*) \quad = \quad \inf_{v \in H^1_0(\Omega)} \mathcal{L}(v, \tau, \mu^*) \quad = \quad \begin{cases} - \, \frac{1}{2} \, \displaystyle\int_\Omega \tau^T A^{-1} \tau \, d\xi \, + \, \langle \, \psi, \, \mu^* \, \rangle_E & ; \quad \tau \in Q^*_{\mu^*} \\ - \, \infty & ; \quad \text{else} \end{cases} .$$

Hereby, the set $Q^*_{\mu^*}$ of admissible vector fields is defined for any measure $\mu^* \in \mathfrak{M}_\psi$ by:

$$Q^*_{\mu^*} \quad := \quad \Big\{ \, \tau \in L^2(\Omega, \mathbb{R}^n) \, \Big| \, \int_\Omega \big\{ \, \tau \nabla v \, - \, f \, v \, \big\} \, d\xi \, = \, \langle \, v, \mu^* \, \rangle_E \ ; \ v \in H^1_0(\Omega) \, \Big\} \ . \qquad (2.17)$$

Combining both results the following bound on the approximation error is inferred from (1.8):

$$\frac{A_0}{2} \, | \, x - x_0 \, |^2_{\Omega,1} \quad \leq \quad F(\nabla x - A^{-1}\sigma) \, + \, F^*(\sigma - \tau) \, + \, \langle \, \sigma - \tau, \nabla x - A^{-1}\sigma \, \rangle_\Omega \, + \, \langle \, x - \psi, \mu^* \, \rangle_E$$

with $\sigma \in L^2(\Omega, \mathbb{R}^n)$ and $\tau \in Q^*_{\mu^*}$ being arbitrary vector fields. Again the device (1.14) may be employed to rid ourselves of the dual pairing. We introduce a parameter $\kappa > 0$ and find:

$$\frac{A_0}{2} \, | \, x - x_0 \, |^2_{\Omega,1} \quad \leq \quad \frac{1+\kappa}{\kappa} \, F(\, \nabla x - A^{-1} \sigma \,) \; + \; (1+\kappa) \, F^*(\sigma - \tau) \; + \; \langle \, x - \psi, \, \mu^* \, \rangle_{_E} \quad .$$

In order to control the right hand side of the above inequality we have to find an upper bound on the expression $F^*(\sigma - \tau)$. We will do this by the very same method we have employed in [41]. That is, we introduce the auxiliary function $\mathcal{Q} : L^2(\Omega, \mathbb{R}^n) \times H^1_0(\Omega) \longrightarrow \mathbb{R}$:

$$\mathcal{Q}(\varsigma, w) \quad := \quad \int\limits_{\Omega} \left\{ \, \varsigma \, \nabla w \; - \; \frac{1}{2} \, \nabla w^T A \, \nabla w \, \right\} \, d\xi \quad . \tag{2.18}$$

We exploit the fact, that we can decompose any vector field $\sigma \in L^2(\Omega, \mathbb{R}^n)$ into a solenoidal component $\sigma_0 \in H_0(\Omega)$ and a second component, which we can compute from a scalar "potential" $s \in H^1_0(\Omega)$. In the case $A \equiv I \in \mathbb{R}^{n \times n}$ such a decomposition is generally known as *Helmholtz splitting* . Let us assume, we have found such a decomposition:

$$\sigma \quad = \quad A \, \nabla s \quad + \quad \sigma_0 \quad .$$

Thanks to the definition of the auxiliary function $\mathcal{Q}$ we now find:

$$\begin{aligned} \sup_{w \in H^1_0(\Omega)} \left\{ \, \inf_{\tau \in H_0(\Omega)} \mathcal{Q}(\sigma - \tau, w) \, \right\} \quad &= \quad \sup_{w \in H^1_0(\Omega)} \int\limits_{\Omega} \left\{ \, \nabla s^T A \, \nabla w \; - \; \frac{1}{2} \, \nabla w^T A \, \nabla w \, \right\} \, d\xi \\ \\ &\geq \quad \frac{1}{2} \, \langle \, \nabla s, A \, \nabla s \, \rangle_{\Omega} \quad \geq \quad \inf_{\tau \in H_0(\Omega)} F^*(\sigma - \tau) \quad . \end{aligned}$$

An upper bound on the right hand side of the above estimate may be obtained by:

$$\begin{aligned} \sup_{w \in H^1_0(\Omega)} \left\{ \, \inf_{\tau \in H_0(\Omega)} \mathcal{Q}(\sigma - \tau, w) \, \right\} \quad &\leq \quad \inf_{\tau \in H_0(\Omega)} \left\{ \, \sup_{\varsigma \in L^2(\Omega, \mathbb{R}^n)} \int\limits_{\Omega} \left\{ \, \nabla s^T A \varsigma \; - \; \frac{1}{2} \, \varsigma^T A \varsigma \, \right\} \, d\xi \, \right\} \\ \\ &= \quad \frac{1}{2} \, \langle \, \nabla s, A \, \nabla s \, \rangle_{\Omega} \quad = \quad F^*(\sigma - \sigma_0) \quad . \end{aligned}$$

On closer inspection of the definition (2.17) we note, that the set $Q^*_{\mu^*}$ of admissible vector fields is an affine space, which may be generated by translating each point in the space $H_0(\Omega)$ of solenoidal vector fields by a fixed element $\hat{\tau} \in Q^*_{\mu^*}$. Substituting σ by $\sigma - \sigma_0 - \hat{\tau}$ we conclude:

$$\inf_{\tau \in Q^*_{\mu^*}} F^*(\sigma - \tau) \quad \leq \quad \sup_{w \in H^1_0(\Omega)} \left\{ \, \inf_{\tau \in Q^*_{\mu}} \mathcal{Q}(\sigma - \tau, w) \, \right\} \quad \leq \quad F^*(\sigma - \hat{\tau}) \qquad ; \qquad \hat{\tau} \in Q^*_{\mu} \quad .$$

As the left hand side of the inequality is independent of $\hat{\tau} \in Q^*_{\mu}$ we may take the infimum with respect to this vector field. Combining the definitions (2.17) and (2.18) we now see:

$$\inf_{\tau \in Q^*_{\mu^*}} F^*(\sigma - \tau) \quad = \quad \sup_{w \in H^1_0(\Omega)} \left\{ \int\limits_{\Omega} \left\{ \, \sigma \, \nabla w \; - \; \frac{1}{2} \, \nabla w^T A \, \nabla w \; - \; f \, w \, \right\} \, d\xi \; - \; \langle \, w, \, \mu^* \, \rangle_{_E} \, \right\} \quad .$$

In order to control the right and side of the above equation we have to introduce two assumptions on the regularity of the data involved: The first assumption $\sigma \in H_{\mathrm{div}}(\Omega)$ is not unduely restrictive, if the data of the variational statement (2.12) warrants a sufficiently regular solution $x_0 \in V_{\psi}$. In the present case our requirements on the data as stated in section 2.2.2 guarantee $x_0 \in H^2(\Omega)$. The second assumption may be more problematic, especially if we allow for thin obstacles $E \neq \Omega$. Henceforth, let us suppose, that $\mu^* \in \mathfrak{M}_{\psi} \cap L^2(\Omega)$ holds. We remark, that the set $\mathfrak{M}_{\psi} \cap L^2(\Omega)$ contains at least the zero function, whence our second condition will not be void, even if E is a set of Lebesgue measure zero. It seems, that no regularity results have been published for elliptic variational inequalities with general thin obstacles. The so called *interior thin obstacle problem*, which may be seen as a transmission problem across an interface separating two subdomains, has

been more intensively researched (see e. g. chapter 11 in [74] and the references cited therein). If the thin obstacle constitutes one part of the boundary $\partial\Omega$, the variational formulation (2.12) is also known as *Signorini problem*. In the scalar case the C^1-regularity of its solution has first been analysed in [72]. In the vector valued case, which corresponds to the contact problem in elastomechanics, regularity results were first obtained in [69,88]. Hölder continuity of the solution and its first derivatives was proved in a three-dimensional setting quite lately [131]. Since little is known of the properties of thin obstacle problems, if $E \subset \Omega$ is an arbitrary compact subset, any attempt at studying such problems in the most general setting is clearly beyond the scope of this work. Hence, let us abide by our incipient requirement, that the set E and the domain Ω coincide - or at least let us assume that E has a non-empty interior. We note:

$$\begin{aligned}\inf_{\tau\in Q^*_{\mu^*}} F^*(\sigma-\tau) &= \sup_{w\in H^1_0(\Omega)} \int_\Omega \Big\{ -w\,(\,\mu^* + f + \nabla\cdot\sigma\,) - \frac{1}{2}\,\nabla w^T A\,\nabla w \Big\}\, d\xi \\ &\le \sup_{w\in H^1_0(\Omega)} \Big\{ \|w\|_\Omega \|\mu^* + f + \nabla\cdot\sigma\|_\Omega - \frac{A_0\,\lambda_0^2}{2}\|w\|^2_\Omega \Big\} = \frac{\|\,\mu^* + f + \nabla\cdot\sigma\,\|^2_\Omega}{2\,A_0\,\lambda_0^2}\,.\end{aligned}$$

Combining this new result with the error estimate we have obtained previously we find:

$$\frac{A_0}{2}\,|\,x-x_0\,|^2_{\Omega,1} \le \frac{1+\kappa}{\kappa}\,F(\,\nabla x - A^{-1}\sigma\,) + \int_\Omega \Big\{ \frac{1+\kappa}{2A_0\lambda_0^2}\,(\,\mu^* + f + \nabla\cdot\sigma\,)^2 + \mu^*(x-\psi) \Big\}\, d\xi$$

with $\mu^* \ge 0$ being an arbitrary square integrable function, whose support is contained within the set E. Since the integrand is simply some function in the space $L^1(\Omega)$, whatever Lagrange multiplier $\mu^* \in L^2(\Omega)\cap\mathfrak{M}_\psi$ we choose, we can calculate the infimum of the right hand side in the above estimate with respect to μ^* by constructing the infimum of the integrand in a point by point fashion. Thus we arrive at the following error a posteriori bound:

$$\begin{aligned}|\,x-x_0\,|^2_{\Omega,1} \le\ & \frac{2}{A_0}\,\frac{1+\kappa}{\kappa}\,F(\,\nabla x - A^{-1}\sigma\,) + \frac{1+\kappa}{A_0^2\lambda_0^2}\,\|\,\nabla\cdot\sigma + f\,\|^2_\Omega \\ & - \frac{1+\kappa}{A_0^2\lambda_0^2}\,\Big\|\,0 \wedge \Big(\,A_0\,\lambda_0^2\,\frac{x-\psi}{1+\kappa} + \nabla\cdot\sigma + f\,\Big)\,\Big\|^2_E\,.\end{aligned} \tag{2.19}$$

Again, the symbol $u\wedge v$ denotes the common greatest lower bound of two functions $u, v \in L^2(\Omega)$.

2.2.6 Preliminary Remarks on the Efficiency

In the two preceding paragraphs, we have developed two a posteriori estimates for the energy error, that are applicable to the obstacle problem as a generic example for variational formulations posed on convex subsets instead of whole linear spaces. While the mathematical technology we have presented in chapter 1 yields a posteriori estimates, that are asymptotically exact for the Laplace problem, this need not necessarily be the case any longer, if we impose constraints on the minimiser. Hence, it will be necessary to examine the estimates (2.15), (2.16) and (2.19) more closely in order to gauge the efficiency of the estimates and evaluate the impact of the constraint on the quality of the error bounds. Let us commence our investigation by observing, that we may not hope to recover the true discretisation error as a matter of principle: As we start our analysis with the abstract error estimate

$$2\;\phi(\,\|x-x_0\|_\Lambda\,) \le J(x,\Lambda x) - J(x_0,\Lambda x_0)$$

the lowest bound on the approximation error we are eventually able to obtain is a function of the right hand side in the above inequality rather than a function of the approximation error itself. The Laplace problem we have studied in the section 2.1 is special in so far, as the approximation error can be expressed in terms of the energy functional:

$$\begin{aligned}2\,\phi(\,\|x-x_0\|_\Lambda\,) &= \frac{1}{2}\,\|\nabla x\|^2_\Omega + \frac{1}{2}\,\|\nabla x_0\|^2_\Omega - \langle\,\nabla x, \nabla x_0\,\rangle_\Omega \\ &= J(x) - J(x_0) + \Big\{\,\langle\,x-x_0, f\,\rangle_\Omega - \langle\,\nabla(x-x_0), \nabla x_0\,\rangle_\Omega\,\Big\} \\ &= J(x) - J(x_0)\quad.\end{aligned}$$

The expression within the pair of braces is the very optimality condition, the stationary point $x_0 \in H_0^1(\Omega)$ has to meet. In the case of the obstacle problem the corresponding condition reads:

$$x_0 \in V_\psi \qquad : \qquad \langle\, \nabla(x - x_0), \nabla x_0 \,\rangle_\Omega \quad \geq \quad \langle\, x - x_0, f \,\rangle_\Omega \qquad ; \qquad x \in V_\psi$$

with the cone $V_\psi \subset H_0^1(\Omega)$ being defined by (2.11). Depending on our choice of the numerical approximation $x \in V_\psi$ the contribution of the first order optimality condition to the above error representation may therefore be positive.

2.2.7 An Analysis of the Error Estimates

In order to understand the impact of our various calculations on the quality of the emerging hypercycle estimates (2.15), (2.16) and (2.19) let us again employ mathematical techniques from the calculus of conjugate duality. They will also enable us to exhibit the very point $\sigma_0 \in L^2(\Omega, \mathbb{R}^n)$ which the possibly sharpest error bound is attained at. We note:

$$\frac{1}{2}\, |\, x^* \,|^2_{\Omega,-1} \quad = \quad \sup_{w \in H_0^1(\Omega)} \left\{\, \langle\, w, x^* \,\rangle_\Omega \,-\, \frac{1}{2}\, \|\nabla w\|^2_\Omega \,\right\} \qquad ; \qquad x^* \in H^{-1}(\Omega) \quad .$$

Exploiting the above identity we can recast the error estimate (2.15) into the following form:

$$\begin{aligned}
M_1 \quad := \quad & \inf_{\sigma \in L^2(\Omega,\mathbb{R}^n)} \left\{\, \frac{2}{A_0}\, \frac{1+\kappa}{\kappa}\, F(\,\nabla x - A^{-1}\sigma\,) \;+\; \frac{\kappa+1}{A_0^2} \inf_{\mu \in \mathfrak{M}_x} |\, \mu + f + \nabla\cdot\,\sigma\,|^2_{\Omega,-1} \,\right\} \\
= \quad & \inf_{\sigma \in L^2(\Omega,\mathbb{R}^n)} \left\{ \quad \frac{2}{A_0}\, \frac{1+\kappa}{\kappa} \sup_{\tau \in L^2(\Omega,\mathbb{R}^n)} \left\{\, \langle\, \nabla x - A^{-1}\sigma, \tau \,\rangle_\Omega \,-\, F^*(\tau) \,\right\} \quad + \right. \\
& \qquad \left. + \quad 2\, \frac{1+\kappa}{A_0^2} \inf_{\mu \in \mathfrak{M}_x} \Big(\sup_{w \in H_0^1(\Omega)} \left\{\, \langle\, w, \mu + f + \nabla\cdot\,\sigma \,\rangle_\Omega \,-\, \frac{1}{2}\, \|\nabla w\|^2_\Omega \,\right\} \Big) \quad \right\} \\
= \quad & \sup_{\tau \in L^2(\Omega,\mathbb{R}^n)} \sup_{w \in \mathfrak{M}_x^*} \left\{ \quad \inf_{\sigma \in L^2(\Omega,\mathbb{R}^n)} \left\{ \quad \frac{2}{A_0^2}\, \frac{1+\kappa}{\kappa}\, \big\langle\, -\,A_0\, A^{-1}\,\tau \,-\, \kappa\,\nabla w,\, \sigma \,\big\rangle_\Omega \,\right\} \quad - \right. \\
& \left. -\; \frac{2}{A_0}\, \frac{1+\kappa}{\kappa}\, F^*(\tau) \,+\, \frac{2}{A_0}\, \frac{1+\kappa}{\kappa}\, \langle\, \nabla x, \tau \,\rangle_\Omega \,-\, \frac{1+\kappa}{A_0^2}\, \|\nabla w\|^2_\Omega \,+\, 2\, \frac{1+\kappa}{A_0^2}\, \langle\, w, f \,\rangle_\Omega \,\right\} .
\end{aligned}$$

Hereby, we have tacitly introduced the *polar* cone $\mathfrak{M}_x^* \subset H_0^1(\Omega)$ associated with the set $\mathfrak{M}_x$ as it has been specified at the end of section 2.2.3. The polar cone $\mathfrak{M}_x^*$ contains all those functions, which yield positive results for any functional from the set $\mathfrak{M}_x$:

$$\mathfrak{M}_x^* \quad := \quad \left\{\, v \in H_0^1(\Omega) \;\middle|\; \langle\, v, x^* \,\rangle_\Omega \;\geq\; 0 \quad ; \quad x^* \in \mathfrak{M}_x \,\right\} \quad .$$

Since the error bound M_1 is defined in terms of a Lagrangian, so to speak, which depends on the variable $\sigma \in L^2(\Omega, \mathbb{R}^n)$ in a linear fashion, a finite value of M_1 can only be attained if

$$\tau \qquad = \qquad -\; \frac{\kappa}{A_0}\; A\,\nabla w \tag{2.20}$$

holds. With a view to the definition of the conjugate functional F^* we therefore conclude:

$$\frac{A_0}{2}\, M_1 \quad = \quad \frac{1+\kappa}{\kappa} \sup_{w \in \mathfrak{M}_x^*} \left\{\, \langle\, w, f \,\rangle_\Omega - \, F(\nabla w) \,-\, \langle\, \nabla x, A\nabla w \,\rangle_\Omega - \, \frac{A_0}{\kappa}\, \frac{1}{2}\, |\, w \,|^2_{\Omega,1} \,\right\} \quad .$$

Let us first consider the special case $A \equiv A_0\, I \in \mathbb{R}^{n\times n}$ which corresponds to the Laplace problem, if we ignore the constraint $x_0 \in V_\psi$. Under this very assumption we may eliminate the functional F in favour of the norm. Thus we find:

$$\frac{A_0}{2}\, M_1 \quad = \quad \frac{1+\kappa}{\kappa} \sup_{w \in \mathfrak{M}_x^*} \left\{\, \langle\, w, f \,\rangle_\Omega - \, A_0\, \langle\, \nabla x, \nabla w \,\rangle_\Omega - \, \frac{A_0}{2}\, \frac{1+\kappa}{\kappa}\, |\, w \,|^2_{\Omega,1} \,\right\} \quad .$$

Since the set $\mathfrak{M}_x^*$ is invariant under scaling with an arbitrary positive parameter we can absorb the factor $1+\kappa^{-1}$ into the function $w \in \mathfrak{M}_x^*$. The resulting equation reads:

$$\frac{A_0}{2}\, M_1 \;=\; \sup_{w\in\mathfrak{M}_x^*}\Big\{\, \langle\, w, f\,\rangle_\Omega - A_0\,\langle\, \nabla x, \nabla w\,\rangle_\Omega - \frac{A_0}{2}\,|\, w\,|^2_{\Omega,-1}\,\Big\} \;=\; J(x) \;-\; \inf_{w\in\mathfrak{M}_x^*} J(w+x)$$

with the energy functional $J : H_0^1(\Omega) \longrightarrow \mathbb{R}$ being defined as in subsection 2.2.5. Thanks to the definition of the set $\mathfrak{M}_x^*$ we can warrant the cone

$$V'_\psi \;:=\; \big\{\, v \in H_0^1(\Omega) \;\big|\; v \geq \psi \text{ on } E\cap\Omega_x \text{ in the sense of } H^1(\Omega)\,\big\} \tag{2.21}$$

to be a dense subset of $\mathfrak{M}_x^* + \{x\}$. Because the obstacle ψ and the approximation $x \in V_\psi$ coincide on the domain Ω_x we conclude, that we can bound the error estimator (2.15) by:

$$\frac{A_0}{2}\, M_1 \;=\; J(x) \;-\; \inf_{w\in V'_\psi} J(w) \quad . \tag{2.22}$$

Depending on the actual shape of the coincidence set Ω_x the error bound (2.15) may therefore be the very best, we can possibly obtain in the light of section 2.2.6. The worst result we can encounter corresponds to the unconstrained case $V'_\psi = H_0^1(\Omega)$ which is implied by $\Omega_x = \emptyset$.

Let us proceed with an investigation of the error estimate (2.16). We will dispense with our requirement that the matrix valued function A be constant and isotropic, since there are no advantages to accrue from such an assumption. We recall, that we have supposed the domain $E \subset \Omega$ to feature a non-empty interior. Therefore, the polar cone of the set $L^2(\Omega)\cap\mathfrak{M}_x$ may be specified as the set of all those square integrable functions which are nonnegative almost everywhere on E. As we will presently see, employing functions of a somewhat higher regularity turns out to be more convenient. Since $H_0^1(\Omega)$ is dense in $L^2(\Omega)$ we may state:

$$\begin{aligned}\inf_{\mu\in L^2(\Omega)\cap\mathfrak{M}_x} \|\,\mu + f + \nabla\cdot\sigma\,\|^2_\Omega \;&=\; \inf_{\mu\in L^2(\Omega)\cap\mathfrak{M}_x}\Big\{\, \sup_{w\in H_0^1(\Omega)}\Big\{\, \langle\, 2\,w, \mu + f + \nabla\cdot\sigma\,\rangle_\Omega - \|w\|^2_\Omega\,\Big\}\Big\} \\ &=\; \sup_{w\in\mathfrak{M}_x^*}\Big\{\, \langle\, 2\,w, f\,\rangle_\Omega - \langle\, 2\,\nabla w, \sigma\,\rangle_\Omega - \|w\|^2_\Omega\,\Big\} \quad .\end{aligned}$$

With the help of the above result let us reformulate the estimate (2.16) in the following manner:

$$\begin{aligned} M_2 \;:=\; &\inf_{\sigma\in H_{\mathrm{div}}(\Omega)}\Big\{\, \frac{2}{A_0}\,\frac{1+\kappa}{\kappa}\, F(\,\nabla x - A^{-1}\sigma\,) \;+\; \frac{\kappa+1}{A_0^2\,\lambda_0^2}\, \inf_{\mu\in L^2(\Omega)\cap\mathfrak{M}_x} \|\,\mu + f + \nabla\cdot\sigma\,\|^2_\Omega\,\Big\} \\ =\; &\inf_{\sigma\in H_{\mathrm{div}}(\Omega)}\Big\{\, \frac{2}{A_0}\,\frac{1+\kappa}{\kappa}\, \sup_{\tau\in L^2(\Omega,\mathbb{R}^n)}\Big\{\, \langle\, \nabla x - A^{-1}\sigma, \tau\,\rangle_\Omega \;-\; F^*(\tau)\,\Big\} \;+ \\ &\qquad +\;\; 2\,\frac{1+\kappa}{A_0^2\,\lambda_0^2}\, \sup_{w\in\mathfrak{M}_x^*}\Big\{\, \langle\, w, f\,\rangle_\Omega - \langle\, \nabla w, \sigma\,\rangle_\Omega \;-\; \frac{1}{2}\,\|w\|^2_\Omega\,\Big\}\,\Big\} \\ =\; &\frac{2}{A_0}\,\frac{1+\kappa}{\kappa}\, \sup_{w\in\mathfrak{M}_x^*}\Big\{\, \langle\, w, f\,\rangle_\Omega - \langle\, \nabla x, A\nabla w\,\rangle_\Omega - F(\nabla w) \;-\; \frac{1}{2}\,\frac{A_0\lambda_0^2}{\kappa}\,\|w\|^2_\Omega\,\Big\} \quad . \end{aligned}$$

With a view to the above equation it seems convenient to introduce a new energy functional $\hat{J}_x : H_0^1(\Omega) \longrightarrow \mathbb{R}$, so we can eventually exhibit the error bound M_2 in terms of a variational statement similar to (2.22). After some elementary computations we find:

$$\hat{J}_x(v) \;:=\; F(\nabla v) \;+\; \frac{A_0\lambda_0^2}{2\,\kappa}\,\|v\|^2_\Omega \;-\; \frac{A_0\lambda_0^2}{\kappa}\,\langle\, v, x\,\rangle_\Omega \;-\; \langle\, v, f\,\rangle_\Omega \quad ; \quad v \in H_0^1(\Omega)\ . \tag{2.23}$$

Using this modified energy functional $\hat{J}_x$ the error estimate M_2 may be represented by:

$$\frac{A_0}{2}\, M_2 \;=\; \frac{1+\kappa}{\kappa}\,\Big\{\, \hat{J}_x(x) \;-\; \inf_{w\in V'_\psi} \hat{J}_x(w)\,\Big\} \quad . \tag{2.24}$$

Next, there is the estimate (2.19) to discuss. Our line of approach will be same we have followed above. We must find the conjugate functional $\phi_w^* : L^2(\Omega) \longrightarrow \mathbb{R}$ associated with the mapping

$$\phi_w(v) \quad := \quad \alpha \, \|v\|_\Omega^2 \; - \; \alpha \, \| \, 0 \, \wedge \, (\, \alpha^{-1} w \, + \, v \,) \, \|_E^2 \qquad ; \qquad v \in L^2(\Omega) \quad .$$

Hereby, the function $w \in L^2(\Omega)$ is supposed to be nonnegative and $\alpha > 0$ denotes a parameter. In order to demonstrate that ϕ_w is convex, let us introduce the mapping $\Theta : \mathbb{R} \times \mathbb{R}_0^+ \longrightarrow \mathbb{R}$:

$$\Theta(\xi, \eta) \quad := \quad \left\{ \begin{array}{lll} -2 \, \eta \, \xi \, - \, \alpha^{-1} \, \eta^2 & ; & \xi \, \leq \, -\alpha^{-1} \eta \\ \alpha \, \xi^2 & ; & \text{else} \end{array} \right. \quad .$$

For any choice of the parameter $\eta \geq 0$ the mapping $\Theta(\cdot, \eta)$ is a closed and convex function in its first argument. Since we can reformulate the functional ϕ_w in the following form

$$\phi_w(v) \quad = \quad \alpha \, \|v\|_{\Omega \setminus E}^2 \; + \; \int\limits_E \Theta(v(\xi), w(\xi)) \; d\xi \qquad ; \qquad v \in L^2(\Omega)$$

we infer that ϕ_w is convex and closed. Furthermore, we note that we may compute the Fenchel conjugate $\phi_w^* : L^2(\Omega) \longrightarrow \mathbb{R}$ according to the formula:

$$\phi_w^*(v) \quad = \quad \frac{1}{4 \, \alpha} \, \|v\|_{\Omega \setminus E}^2 \; + \; \int\limits_E \Theta^*(v(\xi), w(\xi)) \; d\xi \qquad ; \qquad v \in L^2(\Omega)$$

with the conjugate mapping $\Theta^* : \mathbb{R} \times \mathbb{R}_0^+ \longrightarrow \mathbb{R}$ being defined by:

$$\Theta^*(\xi^*, \eta) \quad := \quad \sup_{\xi \in \mathbb{R}} \{ \, \xi \, \xi^* - \, \Theta(\xi, \eta) \, \} \quad = \quad \left\{ \begin{array}{lll} (4\alpha)^{-1} \, |\xi^*|^2 & ; & \xi^* \, \geq \, -2\eta \\ + \, \infty & ; & \text{else} \end{array} \right. \quad .$$

We may define the parameter α by requiring $2 \, A_0 \lambda_0^2 \, \alpha = 1 + \kappa$ and the function w by specifying $2 \, w = x - \psi$. Using these settings we can exhibit the optimal error bound to be derived from the a posteriori estimate (2.19) in the following manner:

$$\begin{aligned} M_3 \quad := \quad & \inf_{\sigma \in H_{\mathrm{div}}(\Omega)} \left\{ \; \frac{2}{A_0} \, \frac{1+\kappa}{\kappa} \; F(\, \nabla x - A^{-1} \sigma \,) \; + \; \frac{2}{A_0} \, \phi_w(\, \nabla \cdot \sigma + f \,) \; \right\} \\ = \quad & \inf_{\sigma \in H_{\mathrm{div}}(\Omega)} \left\{ \; \frac{2}{A_0} \, \frac{1+\kappa}{\kappa} \sup_{\tau \in L^2(\Omega, \mathbb{R}^n)} \left\{ \, \langle \, \nabla x - A^{-1} \sigma, \tau \, \rangle_\Omega \; - \; F^*(\tau) \, \right\} \quad + \right. \\ & \qquad \left. + \qquad \frac{2}{A_0} \sup_{v \in H_0^1(\Omega)} \left\{ \, \langle \, v, f \, \rangle_\Omega - \, \langle \, \nabla v, \sigma \, \rangle_\Omega \; - \; \phi_w^*(v) \, \right\} \; \right\} \\ = \quad & \frac{2}{A_0} \sup_{v \, \in \, V_{\psi - x}} \left\{ \, \langle \, v, f \, \rangle_\Omega - \, \langle \, \nabla x, A \nabla v \, \rangle_\Omega - \, \frac{\kappa}{1+\kappa} \, F(\nabla v) \; - \; \frac{A_0 \lambda_0^2}{2 \, (1+\kappa)} \, \|v\|_\Omega^2 \, \right\} \quad . \end{aligned}$$

By scaling the function v with the reciprocal of $1 + \kappa^{-1}$ we find:

$$\frac{A_0}{2} \, M_3 \quad = \quad \frac{1+\kappa}{\kappa} \sup_{v \, \in \, V_{\psi'}} \left\{ \, \langle \, v, f \, \rangle_\Omega - \, F(\nabla v) \; - \; \langle \, \nabla x, A \nabla v \, \rangle_\Omega - \, \frac{A_0 \lambda_0^2}{2 \, \kappa} \, \| \, v \, \|_\Omega^2 \, \right\} \quad .$$

Hereby, the new cone $V_{\psi'} \subset H_0^1(\Omega)$ is defined in analogy to (2.11) in terms of the scaled function $\psi' := (\psi - x)/(1 + \kappa^{-1})$. With a view to the definition of the functional $\hat{J}_x$ we conclude:

$$\frac{A_0}{2} \, M_3 \quad = \quad \frac{1+\kappa}{\kappa} \; \left\{ \; \hat{J}_x(x) \; - \; \inf_{w \in V_\psi^\kappa} \hat{J}_x(v) \; \right\} \quad . \tag{2.25}$$

The cone V_ψ^κ can be obtained from $V_{\psi'}$ by a simple translation:

$$V_\psi^\kappa \quad := \quad \left\{ \, v \in H_0^1(\Omega) \; \middle| \; v \; \geq \; \frac{\kappa \, \psi + x}{\kappa + 1} \; \text{ on } E \text{ in the sense of } H^1(\Omega) \, \right\} \quad .$$

Finally, let us now abandon our assumption, that the matrix valued function be constant and isotropic. The error bounds M_1 and M_2 feature basically the same structure. Hence, we may introduce a modified energy functional $\tilde{J}_x : H_0^1(\Omega) \longrightarrow \mathbb{R}$ similar to the one $\hat{J}_x$ used in the analysis of the majorant M_2 and thus bound the approximation error by:

$$\frac{A_0}{2}\, M_1 \quad = \quad \frac{1+\kappa}{\kappa} \left\{ \; \tilde{J}_x(x) \; - \; \inf_{w \in V_\psi'} \tilde{J}_x(w) \; \right\} \quad . \tag{2.26}$$

We remark, the cone of admissible functions as defined by (2.21) is the same in both cases. The modified energy functional must account for the L^2-norm being replaced by the H^1-seminorm. Accordingly, the new functional $\tilde{J}$ reads:

$$\tilde{J}_x(v) \quad := \quad F(\nabla v) \; + \; \frac{A_0}{2\,\kappa} \, |\, v \,|_{\Omega,1}^2 \; - \; \frac{A_0}{\kappa} \, \langle \, \nabla v, \nabla x \, \rangle_\Omega \; - \; \langle \, v, f \, \rangle_\Omega \quad ; \quad v \in H_0^1(\Omega) \; .$$

2.2.8 A Comparison of the Error Estimates

In the preceding section we have demonstrated, that the error bounds (2.15), (2.16) and (2.19) can be understood as the dual formulation of minimisation problems involving perturbed energy functionals and modified cones of admissible functions. With a view to their respective definitions we can state for any parameter $\kappa > 0$:

$$V_\psi^\kappa \quad \subseteq \quad V_\psi \quad \subseteq \quad V_\psi' \quad .$$

Since the estimate (2.16) and (2.19) involve the same energy functional $\hat{J}$ we conclude, that the latter error bound is always sharper than the former one. Comparing (2.15) with (2.19) seems to be very difficult. If the matrix valued function A is constant and isotropic, the formula (2.22) shows, that the majorant (2.15) can become as sharp as technically possible. Let us define:

$$M_0 \quad := \quad \frac{2}{A_0} \left\{ \; J(x) \; - \; \inf_{w \in V_\psi} J(w) \; \right\} \quad . \tag{2.27}$$

The above quantity is the best bound on the approximation error we can hope to obtain from any a posteriori error estimator for the variational problem (2.12). We have introduced the set Ω_x of all those points, at which the obstacle function ψ and the numerical solution x of the obstacle problem have come into contact. Let us denote the coincidence set of the obstacle and the analytical solution by Ω_0. We contend, that the hypercycle estimate (2.15) will be optimal, if $\Omega_0 \subseteq \Omega_x$ holds. To prove our claim let us remark, that we may replace in (2.11) the set E by Ω_0 without changing the analytical solution x_0 of the variational problem (2.12). In consequence, we may assume that $\Omega_0 \subseteq \Omega_x$ implies $E \subseteq \Omega_x$ and thence infer $V_\psi = V_\psi'$. We conclude:

$$M_0 \quad = \quad M_1 \quad \leq \quad M_3 \quad \leq \quad M_2$$

if we postulate, that the coincidence set associated with the numerical solution x contains Ω_0. When we consider general matrix valued functions $A \in W^{1,\infty}(\Omega, \mathbb{R}^{n\times n}_{\text{sym}})$, it is no longer obvious how the bounds M_1 and M_3 should be related to one another. We can merely assert:

$$M_0 \quad \leq \quad \max\{\, M_1, M_3 \,\} \quad \leq \quad M_2 \quad .$$

In the limit $\kappa \to \infty$ both auxiliary functionals $\hat{J}_x$ and $\tilde{J}_x$ converge in a point by point manner to the Dirichlet integral J. Furthermore, the cone V_ψ^κ grows with growing κ and exhausts the set V_ψ in the limit $\kappa \to \infty$. To formalise this notion we may state:

$$\bigcup_{\kappa > 0} V_\psi^\kappa \quad = \quad V_\psi \quad .$$

We collect, that the original variational statement of the obstacle problem is recovered in the limit $\kappa \to \infty$. To corroborate our notion let us remark, that for any $\kappa > 0$ the functional $\hat{J}_x$ attains its minimum in the cone V_ψ^κ at the very point x_0, when we choose $x = x_0$. We contend:

Proposition 2.1 *For any value of the perturbation parameter $\kappa > 0$ the solution $x_0 \in V_\psi$ of the primal variational problem (2.12) as it has been detailed in section 2.2.2 coincides with the solution $u_{\kappa,x} \in V_\psi^\kappa$ of the perturbed minimisation problem*

$$\hat{J}_x(v) \quad \underset{v \in V_\psi^\kappa}{\longrightarrow} \quad \min \tag{2.28}$$

with the energy functional $\hat{J}_x : H_0^1(\Omega) \longrightarrow \mathbb{R}$ being defined by (2.23). There is a constant $C_A > 0$, such that for any two admissible functions $x, x' \in V_\psi$ the following a priori estimate holds:

$$\left| u_{\kappa,x} - u_{\kappa,x'} \right|_{\Omega,1} \quad \leq \quad \frac{1}{1+\kappa} \left\{ C_A \left| x - x' \right|_{\Omega,1} + \lambda_0^2 \left| x - x' \right|_{\Omega,-1} \right\} .$$

Proof By introducing the new variable $\tilde{v} := (1+\kappa^{-1})\, v - \kappa^{-1} x$ and dropping all constant expressions, that result from this translation, we derive from the definition (2.23) a new energy functional. After scaling with the factor $(1+\kappa^{-1})^2$ this functional reads:

$$\hat{J}_{\kappa,x}(\tilde{v}) \;:=\; F(\nabla \tilde{v}) + \frac{A_0 \lambda_0^2}{2\,\kappa} \left\| \tilde{v} \right\|_\Omega^2 - \frac{A_0 \lambda_0^2}{\kappa} \left\langle \tilde{v}, x \right\rangle_\Omega - \frac{1+\kappa}{\kappa} \left\langle \tilde{v}, f \right\rangle_\Omega + \frac{1}{\kappa} \left\langle \nabla \tilde{v}, A \nabla x \right\rangle_\Omega$$

with $\tilde{v} \in V_\psi$. By definition $\tilde{v} \in V_\psi$ implies $v \in V_\psi^\kappa$ and vice versa. Let us denote the solution of the auxiliary minimisation problem

$$\hat{J}_{\kappa,x}(\tilde{v}) \quad \underset{\tilde{v} \in V_\psi}{\longrightarrow} \quad \min$$

by $\tilde{u}_{\kappa,x} \in V_\psi$. Since the quadratic part of the energy functional $\hat{J}_{\kappa,x}$ is coercive on $H_0^1(\Omega)$ with some constant, which is bounded from below by A_0 for any choice of the parameter κ, we can easily adapt the classical stability result of Stampacchia and Lions [102]:

$$\sqrt{A_0} \left| \tilde{u}_{\kappa,x} - \tilde{u}_{\kappa,x'} \right|_{\Omega,1} \quad \leq \quad \frac{1}{\kappa} \left\{ \sqrt{A_\infty} \left| x - x' \right|_{\Omega,1} + \sqrt{A_0}\, \lambda_0^2 \left| x - x' \right|_{\Omega,-1} \right\} .$$

Hereby, the constant A_∞ denotes an upper bound on the supremum norm of the matrix valued function A defined in section 2.2.2. The final estimate is now obtained by taking into account the affine transformation we have applied initially:

$$\left| u_{\kappa,x} - u_{\kappa,x'} \right|_{\Omega,1} \quad \leq \quad \frac{\kappa}{1+\kappa} \left| \tilde{u}_{\kappa,x} - \tilde{u}_{\kappa,x'} \right|_{\Omega,1} + \frac{1}{1+\kappa} \left| x - x' \right|_{\Omega,1} .$$

To complete the proof we have to show, that the minimiser of the functional $\hat{J}_{x_0}$ in the cone V_ψ^κ is always x_0. To this end we note, we can decompose the functional into two parts:

$$\hat{J}_{x_0}(v) \quad =: \quad J(v) \quad + \quad J_0(v) \qquad ; \qquad v \in H_0^1(\Omega) .$$

By definition of the analytical solution the Dirichlet integral J will attain its minimum in the cone V_ψ at the point $x_0 \in V_\psi^\kappa$. The second functional $J_0 : H_0^1(\Omega) \longrightarrow \mathbb{R}$ is defined by:

$$J_0(v) \quad = \quad \frac{A_0 \lambda_0^2}{2\,\kappa} \left\| v \right\|_\Omega^2 \quad - \quad \frac{A_0 \lambda_0^2}{\kappa} \left\langle v, x_0 \right\rangle_\Omega \qquad ; \qquad v \in H_0^1(\Omega) .$$

This part of the energy functional $\hat{J}_{x_0}$ will attain its minimum over the whole space $H_0^1(\Omega)$ at the very same point. Consequently, the analytical solution $x_0 \in V_\psi^\kappa$ of problem (2.12) is also the minimiser of the perturbed problem (2.28), if we assume $x = x_0$. □

The above result implies $\left| x_0 - u_{\kappa,x} \right|_{\Omega,1} = \mathcal{O}(\kappa^{-1})$ for any $x \in V_\psi$. To complement our findings let us formulate another proposition which asserts a bound on the efficiency of the estimator M_3:

Proposition 2.2 *If we denote the right hand side of the error estimate (2.19) by M_3 and define M_0 as the smallest possible error majorant we can derive with the help of duality techniques according to formula (2.27) we following efficiency estimate holds for any value of the perturbation parameter $\kappa > 0$ and any choice of an admissible approximation $x \in V_\psi$:*

$$M_3 \quad \leq \quad \frac{1+\kappa}{\kappa} \quad M_0 .$$

Proof Using the same notation we have employed in the proof of the previous proposition we infer from (2.25), that we can write the error estimate M_3 as:

$$\begin{aligned} M_3 &= \frac{2\,(1+\kappa)}{\kappa\,A_0}\left\{\,J(x)\;-\;J(u_{\kappa,x})\,\right\} \;-\; \frac{\lambda_0^2\,(1+\kappa)}{\kappa^2}\;\left\|\,x\;-\;u_{\kappa,x}\,\right\|_{\Omega}^2 \\ &\leq \frac{2\,(1+\kappa)}{\kappa\,A_0}\left\{\,J(x)\;-\;\inf_{v\in V_\psi}\,J(v)\,\right\} \;=\; \frac{1+\kappa}{\kappa}\;M_0 \quad . \end{aligned}$$

□

A closer inspection of the arguments we have used to proof proposition 2.1 and 2.2 shows, that we can derive analogous results for the error majorants M_1 and M_2 as long as we can suppose $\Omega_0 \subseteq \Omega_x$. Unfortunately, there is no reason why such an assumption should be valid. If the coincidence set of numerical solution $x \in V_\psi$ is smaller than the proper coincidence set Ω_0 a systematical error is introduced both into M_1 and M_2, such that an upper bound for these error majorants in terms of M_0 is not even available.

Chapter 3

Discretisation Procedures for Duality Error Majorants in a Finite Element context

In the preceding chapter we have studied a number of applications which may be fit into the theoretical framework for a posteriori error estimates as it has been developed in the first chapter. Yet, we have not elaborated how these estimates may be obtained within the context of some numerical procedure, specifically a finite element computation. Below we will address some of the numerical issues we have to be aware of if we use trial spaces of finite dimension to approximate the solution of the primal problem and to seek a minimiser for the error majorant in.

Occasionally, we will need to refer to a number of general concepts underlying all finite element schemes which rely on *meshes* to set up the basis functions for the trial spaces they employ. For our convenience, we therefore devote the first section to outlining those basic principles. To keep our exposition fully self-contained it would be necessary to complement this section with a discussion how to construct trial spaces X_h of finite dimension which are contained in the very Banach space X, the domain which the operator Λ has been defined on. As the resulting variational formulation

$$F(\Lambda x_h) \; + \; G(x_h) \quad \underset{x_h \in X_h}{\longrightarrow} \quad \inf \quad .$$

involves only a finite number of unknown parameters, the so called *degrees of freedom* of the discrete problem, it is amenable to a numerical treatment. Even though the technology behind finite element computations is interesting in itself, we will not present such a discussion, however: primarily because the literature on finite element computations has grown so vast. While the numerical treatment of the primal formulation can hardly cover new ground, the discretisation of the dual problem may still deserve our deliberation.

In the second section of this chapter we will present a recently discovered [40] and somewhat unusual method to represent the dual variable $y^* \in Y^*$ whose judicious choice determines the significance of the resulting duality error majorant. Albeit we have imposed constraints on the regularity of that variable to ensure, we can actually compute the error bounds, we need not necessarily restrict our attention to conforming discretisation schemes which warrant the proper regularity of the finite element representation y_h^*. The discretisation method for the dual variable we shall discuss belongs to the class of *nonconforming* schemes. As such it may be of limited usefulness if we are only interested in constructing error estimates, that we can evaluate easily. However, if we accept the necessity to introduce bounds on certain expressions we may employ less regular trial functions $y_h^* \in Y$ and thus recover some of the more conventional a posteriori error estimates based on element residuals. More detailed information on the interdependencies between various known error estimators and certain choices for the dual variable $y \in Y$ will be presented in the third section.

3.1 General Remarks on Finite Element Methods

Ever since the invention of the finite element method, which may be dated back to the year 1943 and a paper of R. Courant [53] on continuous, piecewise linear ansatz functions with small support, a large number of different finite element schemes have been proposed for the numerical solution of variational problems. Any attempt to give a comprehensive survey of all the finite elements, which are in use nowadays, would be far beyond the scope of this thesis. If only to introduce our notation, it will prove helpful, nevertheless, to recall those constitutive features, practically all finite elements share. In the first paragraph of this section we will attempt to give an abstract description of the finite element method and supply a classification of the various methods in employ. Along the way we will identify a number of requirements, that are constitutive for finite element schemes. Subsequently, we will show how these requirements can be met by parametral techniques and introduce the concept of *parametric elements*. For such elements, we

will recall a number of well known approximation results. In the third paragraph we will address some of the technical issues, which are related to the evaluation of integrals with the help of quadrature formulae.

3.1.1 An abstract description of finite element schemes

Let us suppose, we want to approximate the solution of some variational problem we may describe in terms of certain integrals over the bounded domain $\Omega \subset \mathbb{R}^n$:

$$\int_\Omega F(v, \nabla v, \dots \nabla^k v) \; d\xi \quad \underset{v \in V}{\longrightarrow} \quad \min \tag{3.1}$$

whereby $V \subset W^{k,p}(\Omega, \mathbb{R}^m)$ denotes the set of admissible functions which we have singled out for example by imposing suitable boundary conditions.

The main idea behind all finite element methods is not to consider said variational formulation on the whole of the domain Ω but rather to define a number $J < \infty$ of compact domains $\Omega_j \subset \mathbb{R}^n$ covering Ω and to numerically solve variational problems of the form

$$\int_{\Omega_j} F(v^j, \nabla v^j, \dots \nabla^k v^j) \; d\xi \quad \underset{v^j \in V_j}{\longrightarrow} \quad \min$$

such that $V_j \subset W^{k,p}(\Omega_j, \mathbb{R}^m)$ is a finite dimensional function space. The collection of patches covering the domain Ω we will henceforth refer to as the *subdivision* $\mathcal{M}_J$ of the computational domain $\Omega_0 := \Omega_1 \cup \dots \cup \Omega_J$, or alternatively as the *mesh*. We would like to remark, that these patches must not overlap one another but on some sets of measure 0. The computational domain Ω_0 may actually differ from Ω along the boundary. For each patch Ω_j a set $\Sigma_j \subset W^{-k,p}(\Omega_j)$ of continuous linear functionals is specified, which are linearly independent, if we consider them as defined on the space V_j only. We will call these functionals the *degrees of freedom* associated with the element Ω_j. Said functionals must be *V_j-unisolvent* in the following sense:

$$\sigma(v) \;=\; \sigma(u) \quad ; \quad \sigma \in \Sigma_j \qquad \Longrightarrow \qquad u \;=\; v$$

for any two functions $u, v \in V_j$. Thus, we ensure the existence of a dual basis $\{v_1^j, \dots v_{i_j}^j\} \subset V_j$ of local *shape functions* which are related to the degrees of freedom associated with Ω_j by:

$$\sigma_j^i(\, v_l^j \,) \;=\; \delta_l^i \qquad ; \qquad i, l \;\in\; \{1, \dots i_j\} \;\; .$$

The number i_j denotes the dimension of V_j respectively the cardinality of $\Sigma_j =: \{\sigma_j^1, \dots \sigma_j^{i_j}\}$. In general, we will be unable to extend the local shape functions, such that their extensions are contained in $W^{k,p}(\Omega_0, \mathbb{R}^m)$. Let us therefore introduce the fragmented space $\mathcal{V}_J$:

$$\mathcal{V}_J \;:=\; \left\{ \; v \in L^p(\Omega_0, \mathbb{R}^m) \;\middle|\; v\,|_{\Omega_j} \in V_j \;\; ; \;\; 1 \le j \le J \; \right\} \;\; .$$

A basis $\mathcal{S}_J$ of its dual space V_J^* can be defined in terms of the local degrees of freedom

$$\mathcal{S}_J \;:=\; \bigcup_{j=1}^{J} \Sigma_j \quad ,$$

if we extend each linear functional $\sigma_j^i \in \Sigma_j$ for any element $v \in \mathcal{V}_J$ with the appointment:

$$\mathrm{supp}(v) \;\cap\; \mathrm{int}(\Omega_j) \;=\; \emptyset \qquad \Longrightarrow \qquad \sigma_j^i(v) \;=\; 0 \quad .$$

Due to the regularity requirements to be imposed on the analytical solution of the variational formulation (3.1) we must complement the discrete variational problem

$$\sum_{j=1}^{J} \int_{\Omega_j} F(v^j, \nabla v^j, \dots \nabla^k v^j) \; d\xi \quad \underset{v \in \mathcal{V}_J}{\longrightarrow} \quad \min \tag{3.2}$$

with a finite number $L_J > 0$ of algebraic constraints, which are of the form:

$$\sum_{j=1}^{J} \sum_{i=1}^{i_j} B_{li}^j \, \sigma_j^i(\, v^j \,) \quad = \quad 0 \qquad ; \qquad 1 \, \leq \, l \, \leq \, L_J \ . \tag{3.3}$$

In this way, we effectually discriminate a subspace $\mathfrak{V}_J \subset \mathcal{V}_J$ of ansatz functions, which meet certain regularity requirements. Boundary conditions can be taken into account by formulating a second set of (possibly nonlinear) equations. For simplicity, let us assume homogeneous boundary conditions either of Dirichlet or of Neumann type, such that we an subsume them under (3.3).

When we restrict our search for the minimiser of the discrete problem to the space $\mathfrak{V}_J$, we no longer need all the functionals from the set $\mathcal{S}_J$. By $\mathfrak{S}_J \subset \operatorname{span} \mathcal{S}_J$ let us denote a basis of the dual $\mathfrak{V}_J^*$; this set defines the global *degrees of freedom* associated with the ansatz $\mathfrak{V}_J$. The dual basis with respect to $\mathfrak{S}_J$ specifies the global *shape functions*. Let $\mathfrak{u} \in \mathfrak{V}_J$ denote any such shape function. To control the numerical complexity of the algorithm we use to solve the discrete variational problem we have to stipulate that $\mathfrak{u}$ have a small support:

$$\# \left\{ \, \Omega_j \in \mathcal{M}_J \ \middle| \ \operatorname{supp} \mathfrak{u} \, \cap \, \operatorname{int}(\Omega_j) \, \neq \, \emptyset \, \right\} \quad < \quad C \ . \tag{3.4}$$

Hereby $C > 0$ designates a constant which may depend on the mesh $\mathcal{M}_J$ but not on our actual choice of $\mathfrak{u}$. If the global shape functions are sufficiently regular, such that $\mathfrak{V}_J \subset W^{k,p}(\Omega, \mathbb{R}^m)$ holds, the discretisation scheme is termed *conforming*. If the algebraic constraints (3.3) prove insufficient to ensure we can evaluate the energy functional of the analytical problem on $\mathfrak{V}_J$ the discretisation is called *nonconforming*.

An alternative approach to solving the discrete variational formulation consists in drawing up a saddle point problem, in which the algebraic constraints (3.3) can be found combined with a set of Lagrange multipliers $\lambda \in \mathbb{R}^{L_J}$. We may extend each local shape function v_i^j outside its respective domain Ω_j by 0, such that we can represent any element $v \in \mathcal{V}_J$ by

$$v \quad = \quad \sum_{j=1}^{J} \sum_{i=1}^{i_j} \, x_j^i \ v_i^j \quad = \quad \sum_{j=1}^{J} \sum_{i=1}^{i_j} \, \sigma_j^i(v) \, v_i^j \tag{3.5}$$

using the appropriate linear factors $x_j^i \in \mathbb{R}$. If N_J denotes the number of local degrees of freedom, we can define a Lagrangian $\mathfrak{L} : \mathbb{R}^{N_J} \times \mathbb{R}^{L_J} \longrightarrow \mathbb{R}$, which has the following structure:

$$\mathfrak{L}(x, \lambda) \quad := \quad \sum_{j=1}^{J} \, Q_j(x_j^1, x_j^2, \ldots x_j^{i_j}) \quad - \quad \sum_{l=1}^{L_J} \sum_{j=1}^{J} \sum_{i=1}^{i_j} \, \lambda^l \, B_{li}^j \, x_j^i \quad .$$

The nonlinear functionals $Q_j : \mathbb{R}^{i_j} \longrightarrow \mathbb{R}$ depend but on a very small number of variables, whence it is comparatively simple to find the minimiser of $\mathfrak{L}$ with respect to its first argument, once the Lagrange multipliers $\lambda \in \mathbb{R}^{L_J}$ have been fixed. In consequence, an algorithm of Uzawa type suggests itself to locate the saddle point. To the best of our knowledge such an algorithm has not been analysed as yet. However, it should be straightforward to adapt the results found in [27, 46] for quadratic forms Q_j to more general situations. A finite element scheme which is geared towards computing the Lagrange multipliers $\lambda \in \mathbb{R}^{L_J}$ is termed a *hybrid* method.

3.1.2 A note on parametric finite elements

In the previous subsection we have outlined the general concept of finite element methods. Though we have not mentioned it at the time, our exposition has been biased towards those finite element schemes, which rely on the proper construction of the mesh for the accuracy of the numerical solution. Such schemes are usually referred to as *h-methods*, while schemes which enrich the local trial spaces V_j to improve the quality of the approximation, are generally termed *p-methods*. The rationale behind this nomenclature will become apparent below, when we discuss practical ways how to define finite elements that warrant certain approximation properties.

Let us assume, that all patches $\Omega_j \in \mathcal{M}_J$ in our mesh have more or less the same shape. We will give a precise meaning to this condition in definition 3.1. Let us furthermore suppose, that

each local ansatz V_j contains a full polynomial space of order q. We note, that the right hand side of (3.5) defines a projection $\Pi : W^{k,p}(\Omega_0, \mathbb{R}^n) \longrightarrow \mathcal{V}_J$. Hence, the operator Π maps all piecewise polynomial functions from the space $\mathcal{V}_J$ with a degree less or equal to q onto themselves. Due to our assumptions we can infer from well known theoretical results (see e. g. [49, 50, 59]) on interpolation operators in Sobolev spaces, that the local interpolation error on some patch Ω_j is uniformly bounded by its volume $|\Omega_j|$. For any $q' \leq q < k$ we find:

$$| v - \Pi v |_{p,\Omega_j,q'} \quad \leq \quad C \sqrt[n]{|\Omega_j|}^{\,1+q-q'} \, \| \nabla v \|_{p,\Omega_j,q} \qquad ; \qquad v \in W^{k,p}(\Omega_0, \mathbb{R}^m) \ .$$

A convenient method of constructing meshes, which warrant a local error estimate of the above form, relies on applying various maps $T_j : \mathbb{R}^n \longrightarrow \mathbb{R}^n$ to a fixed *reference element* $\Omega_{\text{ref}} \subset \mathbb{R}^n$, such that the resulting images $\Omega_j = T_j(\Omega_{\text{ref}})$ seamlessly fit together. To be more specific let us formalise the concept of *parametric elements*:

Definition 3.1 *Let $\{\alpha_1, \dots \alpha_K\} \subset \mathbb{R}^n$ denote a set of points and Ω_α the interior of their convex hull. We suppose there are K functions $\{\vartheta_1, \dots \vartheta_K\} \subset W^{k,\infty}(\hat{\Omega})$, such that*

$$\vartheta_i(\alpha_j) \quad = \quad \delta_{ij} \qquad ; \qquad i, j \ \in \ \{1, \dots K\}$$

is satisfied. By $\{a_1, \dots a_K\} \subset \mathbb{R}^n$ we denote a second set of points. The interior of its convex hull we abbreviate Ω_a^c. To describe the geometry of Ω_a^c we introduce two quantities: The radius ρ of the largest sphere, that can be inscribed into Ω_a^c, and the diameter h of this patch:

$$\begin{aligned} \rho \quad &:= \quad \sup\{\, r \geq 0 \ \big| \ x \in \Omega_a^c \ : \ B_r(x) \subset \Omega_a^c \,\} \\ h \quad &:= \quad \max_{1 \leq i,j \leq K} \| a_i - a_j \|_2 \quad . \end{aligned} \tag{3.6}$$

Employing this second set of points we may now define a mapping $T_a \in W^{k,\infty}(\Omega_\alpha, \mathbb{R}^n)$ by:

$$T_a(\xi) \quad := \quad \sum_{\kappa=1}^{K} \vartheta_\kappa(\xi) \ a_\kappa \qquad ; \qquad \xi \ \in \ \Omega_\alpha \ . \tag{3.7}$$

The image of Ω_α under the action of T_a can be invested with an ansatz space and appropriate degrees of freedom to form a finite element $\{\Omega_a, \Sigma, V_a\}$. This object we will call a parametric finite element of degree q, if the following conditions are met:

1.) *The mapping $T_a : \Omega_\alpha \longrightarrow \Omega_a$ has a globally defined inverse map $T_\alpha \in W^{k,\infty}(\Omega_a, \mathbb{R}^n)$.*

2.) *There exists a constant $C > 0$, which is independent of our particular choice of the set $\{a_1, \dots a_K\}$, such that the following estimates hold:*

$$\begin{aligned} | T_a |_{\infty,\Omega_\alpha,l} \quad &\leq \quad C \ h^l \qquad ; \qquad l \ \in \ \{1, \dots q\} \\ | T_\alpha |_{\infty,\Omega_a,l} \quad &\leq \quad C \ \rho^{-1} \qquad ; \qquad l \ \in \ \{1, \dots q\} \ . \end{aligned}$$

3.) *We find $V_a = \{\, u \circ T_\alpha \,|\, u \in V_\alpha \,\}$, where $V_\alpha \subset W^{k,p}(\Omega_\alpha, \mathbb{R}^m)$ is a finite dimensional function space, that contains all polynomials of degree less or equal to q.*

Remark 3.1 The definition 3.1 comprises the concepts of *isoparametric* and *subparametric* finite elements, as such elements are characterised by the additional requirement: $\{\vartheta_1, \dots \vartheta_K\} \subseteq V_\alpha$. Whether we are actually able to cover any given domain with parametric elements, will depend on our choice of the set $\{\alpha_1, \dots \alpha_K\}$ as well as of the functions $\{\vartheta_1, \dots \vartheta_K\}$. One way to guarantee, that neighbouring parametric elements fit together, consists in using polynomials of small degree to define the maps (3.7) and placing a sufficiently large number of the sampling points $\{\alpha_1, \dots \alpha_K\}$ on each part of the boundary $\partial\Omega_\alpha$.

For a parametric element of degree q in the sense of definition 3.1 the local interpolation error of the canonical finite element interpolation operator Π reads (see e. g. [48]):

$$| v - \Pi v |_{p,\Omega_a,q'} \quad \leq \quad C(\Omega_\alpha) \, \frac{h^{\,q+1}}{\rho^{\,q'}} \, \| \nabla v \|_{p,\Omega_a,q}$$

for any function $v \in W^{k,p}(\Omega_0, \mathbb{R}^n)$. Hereby, h and ρ denote the geometry parameters of the patch Ω_a as they are specified in (3.6). We conclude, a global estimate of the interpolation error in terms of the maximal element diameter will only be available, if the quotient h/ρ stays uniformly bounded from above. In effect, we must ensure the elements cannot become too flat:

Definition 3.2 *A mesh $\mathcal{M}_J$ which is composed of parametric elements in the sense of definition 3.1 is called (shape) regular, if there is a constant $C_J > 0$, such that the parameters ρ_j and h_j as specified in (3.6) meet the condition:*

$$\frac{h_j}{\rho_j} \;\leq\; C_J \qquad ; \qquad \Omega_j \;\in\; \mathcal{M}_J \qquad .$$

If some shape regular mesh $\mathcal{M}_J$ is given, that consists of parametric elements of degree q, the global approximation properties of the interpolation operator Π can indeed be described in terms of the largest diameter h, that is found among all patches $\Omega_j \in \mathcal{M}_J$:

$$h \quad := \quad \max_{\Omega_j \in \mathcal{M}_J} \; h_j \qquad . \tag{3.8}$$

In general this *mesh parameter* h is used as in index instead of J, that is the number of patches, to qualify the mesh. Since the image space of the interpolation operator Π need not necessarily be contained in $W^{k,p}(\Omega, \mathbb{R}^n)$ we obtain the following global error estimate:

$$\Big(\sum_{j=1}^{J} \| v_j \,-\, \Pi\, v_j \|^p_{p,\Omega_j,q'} \Big)^{1/p} \quad \leq \quad C(\Omega_\alpha)\, C_J^{q'} \;\; h^{1+q-q'} \;\; \| \nabla v \|_{p,\Omega_0,q} \tag{3.9}$$

provided the function v is sufficiently regular. The above bound need not involve the full norm of the gradient, if the parameterisations is based on polynomial mappings $\{\vartheta_1, \ldots, \vartheta_K\} \subseteq P^n_l \subseteq V_\alpha$ with a degree l not larger than that of the element itself. In that case we find:

$$\Big(\sum_{j=1}^{J} \| v_j \,-\, \Pi\, v_j \|^p_{p,\Omega_j,q'} \Big)^{1/p} \quad \leq \quad C(\Omega_\alpha)\, C_J^{q'} \;\; h^{1+q-q'} \;\; \| \nabla^{q-l+2} v \|_{p,\Omega_0,l-1} \quad .$$

Remark 3.2 We have seen, the local approximation properties of the canonical finite element interpolation $\Pi : W^{p,k}(\Omega_0, \mathbb{R}^n) \longrightarrow \mathcal{V}_J$ depend on the geometry of the patch Ω_j and on the degree q of the polynomial space P^n_q contained in the local ansatz V_j. Since the accuracy of the numerical solution to the problem (3.1) is limited by the accuracy of the finite element interpolation of the analytic solution, the quality of the numerical solution may be improved either by local mesh refinement or by augmenting the local ansatz spaces. The mesh parameter defined by (3.8) is usually denoted by h, whence the first type of approach is termed h-method. The latter type of approach to controlling the approximation error is termed p-method to indicate, the degree p (in our notation q) of the local ansatz is adapted to the required accuracy.

3.1.3 On the Ramifications of the Numerical Cubature

Throughout this text we generally ignore the question, how to compute the various integrals that form a constitutive part of the duality error majorants we have developed in chapter 2. From a theoretical point of view such unconcern for the actual implementation of a numerical scheme may seem tolerable, from a practical point of view redressing any technical difficulties can prove critical for the success of the scheme.

Unless the function F introduced in (3.1) is a rather simple polynomial it is inevitable to employ a numerical cubature to approximate those integrals, which are the building blocks of the discrete variational formulation. Depending on our discretisation scheme we may furthermore want to evaluate certain integrals in order to assemble the tensor B, which is used in (3.3) to describe admissibility constraints on the local shape functions. Hence, the use of a numerical cubature does not only affect the computation of a posteriori error estimates for the numerical solution of the problem (3.1) but also the very method, by which we hope to obtain this solution. In the following we will not discuss in depth how the numerical solution is affected by the cubature:

for more detailed information on that topic we refer to [70, 79, 80]. We will, however, address some of the additional requirements on the finite element discretisation which result from the use of quadrature formulae.

Parametric finite elements as they have been described in the previous subsection are well suited for numerical integration schemes, since their definition allows for the generic construction of quadrature rules $E_j : C^0(\Omega_j) \longrightarrow \mathbb{R}$ for each patch $\Omega_j \in \mathcal{M}_J$. The underlying idea behind the design of such quadrature formulae is to define a cubature on the reference element and to lift it onto the patch Ω_j with the help of the mapping $T_a : \Omega_\alpha \longrightarrow \Omega_a$:

$$E_j(v) \quad := \quad \sum_{l=1}^{L} \Theta_l \ \det(\nabla T_a)(x_l) \ v(\, T_a(x_l)\,) \qquad ; \qquad v \in C^0(\Omega_j) \ .$$

Hereby, the set $\{x_1, \ldots x_L\} \subset \Omega_\alpha$ denotes the collocation points of the cubature rule within the reference element while $\{\Theta_1, \ldots \Theta_L\} \subset \mathbb{R}$ designates the corresponding set of weights. Obviously, for such a numerical integration scheme to be well defined we must warrant that the Jacobian of each mapping $T_a : \Omega_\alpha \longrightarrow \Omega_a$ is a continuous function. Moreover, each local ansatz V_j must be contained at least in $C^m(\Omega_j, \mathbb{R}^n)$ to allow for an evaluation of the locally defined functionals in (3.2). The fully discretised variational formulation reads:

$$H_J(v) \quad := \quad \sum_{j=1}^{J} E_j\big(\, F(v^j, \nabla v^j, \ldots \nabla^k v^j)\,\big) \qquad \underset{v \in \mathfrak{V}_J}{\longrightarrow} \qquad \min \ .$$

For simplicity let us suppose, that the variational formulation (3.1) be uniformly convex. Our assumption will ensure the problem is well posed. Clearly, the discrete problem (3.2) is also well defined provided the finite element scheme is conforming. If the discretisation is nonconforming we must ensure by suitable constraints on the degrees of freedom that a unique minimiser can be found in the trial space $\mathfrak{V}_J$. However, the situation is much more involved, when we replace the integrals by a numerical cubature rule. Central at least to the study of a posteriori error estimates in the spirit of chapter 1 is the question what happens to the so called *exact modulus of convexity*, which may be defined by:

$$\mu_J(t) \quad := \quad \inf_{u \in S_t(v)} \ \inf_{0<\alpha<1} \ \frac{\alpha\, H_J(u) \ + \ (1-\alpha)\, H_J(v) \ - \ H_J(\,\alpha\, u + (1-\alpha)\, v\,)}{\alpha\,(1-\alpha)} \qquad ; \qquad t \ \geq \ 0$$

if we denote by $S_t(v) \subset \mathfrak{V}_J$ the sphere of radius t around the centre point $v \in \mathfrak{V}_J$. The left hand side of (3.9) may thereby provide an appropriate norm for the ansatz $\mathfrak{V}_J$. As far as the original statement (3.1) of the variational problem is concerned we can employ the modulus of convexity μ as a forcing function and bound the approximation error as we have done in section 1.2.2 with $2\,\phi$ being replaced by μ. This very function μ is known [139] to obey the following inequality for any point $t \geq 0$ within its effective domain:

$$\mu(c\,t) \quad \geq \quad c^{\,2}\, \mu(t) \qquad ; \qquad c \ \geq \ 1 \quad . \tag{3.10}$$

Furthermore, we find $\mu(t) > 0$ for any $t > 0$. Unfortunately, the function μ_J as it has been defined above need not necessarily inherit these properties. Only by a judicious choice of the quadrature formulae E_j we can ensure the uniform convexity of the discrete functional H_J. Since μ_J may depend on the geometry of the mesh, we must furthermore warrant, that there is an uniform lower bound μ_∞ on the modulus of convexity which satisfies both (3.10) and $\mu_\infty(t) > 0$ for any $t > 0$. Such a bound shall also be termed *modulus of convexity*. No general principle is known to us by which we could tailor cubature schemes to the specific requirements of the variational problem under consideration. However, in the special case that (3.1) represents basically a bilinear form on the extended domain Ω_0 the following result is available (see e. g. section 4.4 in [48]):

Proposition 3.1 *Let $\mathcal{M}_h$ denote a shape regular decomposition of the domain Ω_0 with the mesh parameter h defined by (3.8). The mesh consists of parametric elements in the sense of definition 3.1 defined with the help of the reference patch $\Omega_\alpha \subset \mathbb{R}^n$ and a finite dimensional function space*

with the property $P_{l'}^m \subseteq V_\alpha \subseteq P_l^m$. *Let* $E_\alpha : C^0(\hat{\Omega}) \longrightarrow \mathbb{R}$ *denote a cubature rule for the reference patch, which is exact on the polynomial space* P_{2l-2} *and satisfies:*

$$\sum_{i=1}^{m} \sum_{|\alpha|=k} E_\alpha\Big(\frac{\partial^\alpha v_i}{\partial x^\alpha} \frac{\partial^\alpha v_i}{\partial x^\alpha} \Big) \quad \geq \quad C \mid v \mid^2_{\Omega_\alpha, k} \qquad ; \qquad v \in V_\alpha \ . \tag{3.11}$$

For each patch $\Omega_j \in \mathcal{M}_h$ *a numerical integration scheme* $E_j : C^0(\Omega_j) \longrightarrow \mathbb{R}$ *is assigned by:*

$$E_j(v) \quad := \quad E_\alpha\big(\det(\nabla T_j)\, v \circ T_j \big) \qquad ; \qquad v \in C^0(\Omega_j)$$

with the mapping $T_j : \Omega_\alpha \longrightarrow \Omega_j$ *being specified as in definition 3.1. The bilinear form*

$$Q_J(v,u) \quad := \sum_{|\alpha|=|\beta|=k} \sum_{i,h=1}^{m} \sum_{j=1}^{J} E_j\Big(M_{\alpha\beta}^{ij} \frac{\partial^\alpha v_i}{\partial x^\alpha} \frac{\partial^\beta u_j}{\partial x^\beta} \Big) \qquad ; \qquad v, u \in C^k(\Omega_0, \mathbb{R}^m)$$

which is defined with the help of the positive definite tensor $M_{\alpha\beta}^{ij} \in C^0(\Omega_0)$ *is uniformly convex on the space* $\mathfrak{V}_J$, *provided either of the these conditions is met: a) The elliptic operator which corresponds to the bilinear form* Q_J *is of second order, that means:* $k = 1$. *b) All the mappings* $T_j : \Omega_\alpha \longrightarrow \Omega_j$ *are affine. In both of these cases a modulus of convexity can be specified, that is independent of the mesh parameter* h.

Remark 3.3 To the best of our knowledge quadrature formulae for parametric elements have not been studied as yet under the assumption, the elliptic operator which corresponds to the bilinear form Q_J is of higher than second order. Since the mapping of higher order derivatives to and fro the reference patch involves certain lower order contributions scaled by higher derivatives of the maps $T_a : \Omega_\alpha \longrightarrow \Omega_a$, the uniform convexity of the quadratic form $Q_J(v,v)$ for $v \in \mathfrak{V}_J$ is in this case no longer an immediate consequence of (3.11). If the coercivity of the quadratic form can be established at all, we must expect our result to be of an asymptotical nature with $h \to 0$ and therefore valid only on a sufficiently fine mesh.

Remark 3.4 Conventional a posteriori error estimators (see e. g. [5, 137] for an overview) which rely on the evaluation of local residuals are derived under the assumption that the numerical solution is in some sense the best possible approximation to the analytical solution among all functions within the trial space. This very feature of the finite element solution, usually termed *Galerkin orthogonality*, is violated if we utilise inexact cubature schemes. We conclude, that the mathematical foundations of residual based error estimators are undermined, whenever numerical quadrature formulae are employed. The difficulties which stem from the numerical cubature lie elsewhere in the case of duality based a posteriori error estimates: Since these estimates can be obtained without any special assumptions on the nature of the primal approximation x except for its admissibility $x \in dom\, G$, only the very evaluation of the error majorants is impinged upon. Hence, we may wave the effects of a numerical cubature as a higher order perturbation if the data has enough regularity to allow for appropriately accurate integration schemes.

3.2 Discretisation Methods for the Dual Formulation

Under the assumption that A is a positive definite and uniformly bounded matrix function an elliptic boundary value problem of the form

$$\nabla\cdot\big(A^{-1} \nabla u \big) \;=\; f \qquad \text{s. t.:} \qquad u \,|_{\,\partial\Omega} \;=\; u_0 \tag{3.12}$$

defined on some domain $\Omega \subset \mathbb{R}^n$ can be solved by minimising its associated energy integral $J : H^1(\Omega) \longrightarrow \mathbb{R}$ defined by:

$$J(v) \quad := \quad \int_\Omega \Big\{ \frac{1}{2} (\nabla v)^T A^{-1} \nabla v \;+\; f\, v \Big\}\, dx \qquad ; \qquad v \in H^1(\Omega) \ .$$

Alternatively, a complementary energy principle may be invoked. Formally, such a procedure can be described by the definition of an auxiliary variable $\sigma := A^{-1}\nabla u$, such that (3.12) is replaced by a system of equations both of which are of first order:

$$\nabla \cdot \sigma \;=\; f \quad , \qquad A\,\sigma \;=\; \nabla\, u \quad . \tag{3.13}$$

Solving the above system is equivalent to minimising the complementary energy functional

$$J^*(\sigma) \;:=\; \frac{1}{2}\int\limits_{\Omega} \sigma^T A\,\sigma \; dx \;-\; \oint\limits_{\partial\Omega} u_0\,\sigma_n \; ds \qquad ; \qquad \sigma \in H_{\mathrm{div}}(\Omega)$$

in the affine space $Q^* := \{\, \tau \in H_{\mathrm{div}}(\Omega) \,|\, \nabla\cdot\tau = f \,\}$ of admissible vector fields. The energy integral $J : H^1(\Omega) \longrightarrow \mathbb{R}$ and its complementary energy $J^* : H_{\mathrm{div}}(\Omega) \longrightarrow \mathbb{R}$ are related by:

$$-\inf_{\tau\in Q^*} J^*(\tau) \;\leq\; J(u+u_0) \qquad ; \qquad u \,\in\, H_0^1(\Omega) \;. \tag{3.14}$$

Hence, dual mixed discretisation schemes for the system (3.13) provide, at least in principle, a mechanism to estimate the approximation error by monitoring the complementary gap. However, conventional mixed finite element methods for (3.13) yield an approximation $u_h \in L^2(\Omega)$ which is not, speaking in mechanical terms, kinematically admissible. Moreover, the requirement $\sigma_h \in Q^*$ will be met only approximately in all but exceptional cases.

Unfortunately, the numerical technology which is suitable for treating the primal formulation differs significantly from the technology that is appropriate for a dual or dual mixed formulation. Differences pertain to many algorithmic features of the finite element engine: grid handling in general, particularly local grid refinement procedures, setup routines for shape functions and support for data interpolation procedures to name but a few of such features. It seems natural to investigate, whether it is possible to solve an elliptic boundary value problem stated in its dual formulation by means of a finite element package geared towards dealing with the primal formulation. As a tool to facilitate such an "abuse" hybrid mixed discretisation schemes will be considered below. To the best of our knowledge the hybridisation of a mixed ansatz was first proposed by Fraeijs de Veubeke [71] with a view to lifting the saddle point character of (3.13). By enforcing continuity constraints on the dual variables only in a weak sense he found it possible to transform (3.13) into a positive definite system for the Lagrange multipliers of these very constraints (see also subsection 3.1.1). Though the new variables are defined only on the skeleton of the grid, in some cases [9, 107] his procedure was discovered to be equivalent to solving the primal problem with the help of certain nonconforming finite element trial spaces. Ever since various authors (e. g. [8, 31]) have researched and exploited this equivalence.

In the following we will present another approach to the definition of Lagrange multipliers for the continuity constraint $\sigma_h \in H_{\mathrm{div}}(\Omega)$, which we believe to be novel. Employing a conventional conforming trial space we introduce these multipliers as traces of certain shape functions. The resulting discretisation for the dual variable will not yield a conforming approximation σ_h. The Lagrange multipliers, however, naturally extend to the whole of the computational domain, where they provide conforming approximations to the primal variables. We may argue, that our scheme complements the established equivalence between conforming dual and nonconforming primal methods. We will show, that an optimal asymptotic rate of convergence can be achieved for both primal and dual variables. Our exposition will be organised in the following way: The variational problem in its dual mixed formulation is presented in the first subsection. In the next paragraph the trial spaces are specified, that are employed in defining the algebraic problem. Existence and uniqueness results for its solution are obtained. Paragraph 3.2.3 contains the main result of this section: a proof of convergence. The subsection 3.2.4 is devoted to post processing techniques for the Lagrange multipliers of the continuity constraints that improve on the rates of convergence established in Section 3.2.3. Numerical experiments will be reported in a dedicated section in chapter 5.

3.2.1 Statement of the Variational Problem

In the following let us assume, that $\Omega \subset \mathbb{R}^n$ is a bounded, convex polytope with $n \in \{2, 3\}$. By $\mathcal{M}_h$ we denote a simplicial decomposition of the domain Ω, such that the intersection of two

elements Ω_i and $\Omega_{i'}$ is again a simplex of lower dimension. (This includes the case that $\Omega_i \cap \Omega_{i'}$ contains only a vertex.) The mesh parameter $h > 0$ is assigned in the usual way to bound the diameter of these simplices (see e. g. §3.1 in [48] or subsection 3.1.2). Since we want to apply a hybrid discretisation scheme to the dual formulation of the problem (3.12) we must account for test functions which satisfy no global regularity constraints. Accordingly, let us introduce the broken spaces $\mathcal{H}^m_{\mathrm{div}}(\mathcal{M}_h)$:

$$\mathcal{H}^m_{\mathrm{div}}(\mathcal{M}_h) \;:=\; \left\{\, \tau \in L^2(\Omega,\mathbb{R}^n) \;\middle|\; \tau|_{\Omega_i} \in H^m_{\mathrm{div}}(\Omega_i) \;;\; \Omega_i \in \mathcal{M}_h \,\right\} \quad .$$

In an analogous way these spaces are equipped with the norms $\|\!|\cdot|\!\|_m$, $\|\cdot\|_m$ and the corresponding seminorms $|\!|\cdot|\!|_m$, $|\cdot|_m$. The norm $\|\!|\cdot|\!\|_m$ for instance is defined by:

$$\|\!|\, \sigma \,|\!\|^2_m \;:=\; \sum_{\Omega_i \in \mathcal{M}_h} \|\!|\, \sigma \,|\!\|^2_{\Omega_i,m} \qquad ; \qquad \sigma \in \mathcal{H}^m_{\mathrm{div}}(\mathcal{M}_h) \quad .$$

Three bilinear forms a, b and c are necessary to specify the hybrid, dual mixed formulation:

$$\begin{aligned} a(\tau,\sigma) \;&:=\; \sum_{\Omega_i \in \mathcal{M}_h} \big(\, \tau \,,\, A\cdot\sigma \,\big)_{\Omega_i} \qquad ; \qquad \sigma\,,\,\tau \;\in\; \mathcal{H}_{\mathrm{div}}(\mathcal{M}_h) \;\;, \\ b(u,\sigma) \;&:=\; \sum_{\Omega_i \in \mathcal{M}_h} \big(\, u \,,\, \nabla\cdot\sigma \,\big)_{\Omega_i} \qquad ; \qquad \sigma \in \mathcal{H}_{\mathrm{div}}(\mathcal{M}_h)\,,\; u \in L^2(\Omega) \quad . \end{aligned}$$

The tensor $A \in W^{1,\infty}(\Omega,\mathbb{R}^{n\times n})$ need not necessarily be symmetric. However, we will impose the usual ellipticity condition:

$$A_0 > 0 \quad : \quad \sum_{i=1}^n \sum_{j=1}^n A_{ij}(x)\,\xi_i\,\xi_j \;\geq\; A_0\,\|\xi\|^2 \quad ; \quad \xi \in \mathbb{R}^n\,,\; x \in \Omega \quad . \tag{3.15}$$

The bilinear form $c : H^1_0(\Omega) \times \mathcal{H}_{\mathrm{div}}(\mathcal{M}_h) \longrightarrow \mathbb{R}$ serves as a means to enforce constraints on the global regularity of the vector fields from the space $\mathcal{H}_{\mathrm{div}}(\mathcal{M}_h)$. For any vector field $\tau \in \mathcal{H}_{\mathrm{div}}(\mathcal{M}_h)$ the following equivalence is stipulated:

$$\sup_{z\in H^1_0(\Omega)} c(z,\tau) \;=\; 0 \qquad \Leftrightarrow \qquad \tau \;\in\; H_{\mathrm{div}}(\Omega) \quad . \tag{3.16}$$

In abstract terms the hybrid, dual mixed variational problem now reads:

Problem H *Given some continuous linear functional $\phi : \mathcal{H}_{\mathrm{div}}(\mathcal{M}_h) \longrightarrow \mathbb{R}$ and some function $f \in L^2(\Omega)$, find a triple $\{\sigma,u,w\} \in H_{\mathrm{div}}(\Omega) \times L^2(\Omega) \times H^1_0(\Omega)$, such that for any set of test functions $\{\tau,y,z\} \in \mathcal{H}_{\mathrm{div}}(\mathcal{M}_h) \times L^2(\Omega) \times H^1_0(\Omega)$:*

$$\left|\begin{array}{lllllll} a(\tau,\sigma) & + & b(u,\tau) & - & c(w,\tau) & = & \phi(\tau) \\ b(y,\sigma) & & & & & = & (f,y)_\Omega \\ c(z,\sigma) & & & & & = & 0 \end{array}\right| \quad .$$

One possible choice for the form c is presented below. The proof that this choice does meet the requirement (3.16) will be supplied in proposition 3.2:

$$c(z,\tau) \;:=\; \sum_{T\in\mathcal{M}_h} \Big\{ \big(z,\nabla\cdot\tau\big)_T + \big(\nabla z,\tau\big)_T \Big\} \qquad ; \qquad z \in H^1_0(\Omega)\,,\; \tau \in \mathcal{H}_{\mathrm{div}}(\mathcal{M}_h)\,. \tag{3.17}$$

Proposition 3.2 *With the bilinear form c specified by (3.17) problem **H** is well posed.*

Proof For $v \in L^2(\Omega)$ let $\omega \in H^1_0(\Omega)$ denote the solution of the equation $\Delta\omega = v$. The vector field $\tau := \nabla\omega$ is contained in $H_{\mathrm{div}}(\Omega)$. Thanks to the elliptic regularity of the auxiliary problem the following estimate holds:

$$\|\!|\, \tau \,|\!\|^2 \;=\; \|\!|\, \tau \,|\!\|^2_\Omega \;=\; \|\, \nabla\omega \,\|^2_\Omega \;+\; \|\, v \,\|^2_\Omega \;\leq\; (\,1 + C\,)\,\|\, v \,\|^2_\Omega \quad .$$

Due to $b(v,\tau) = \|v\|_\Omega^2$ this estimate demonstrates, that b satisfies a compatibility condition, widely known as *Ladyzhenskaya-Babuška-Brezzi* condition:

$$\beta_0 > 0 \quad : \quad \sup_{\tau \in \mathcal{H}_{\rm div}(\mathcal{M}_h)} \frac{b(y,\tau)}{\|\!|\, \tau \,|\!\|} \quad \geq \quad \beta_0 \, \| \, y \, \|_\Omega \quad ; \quad y \in L^2(\Omega) \quad . \tag{3.18}$$

Thanks to (3.15) the first bilinear form a is coercive in the following sense:

$$a(\sigma,\sigma) \quad \geq \quad A_0 \ \|\!|\, \sigma \,|\!\|^2 \qquad ; \qquad \sigma \ \in \ B$$

with the set $B \subset \mathcal{H}_{\rm div}(\mathcal{M}_h)$ of solenoidal vector fields being defined by:

$$B \ := \ \left\{ \, \tau \in \mathcal{H}_{\rm div}(\mathcal{M}_h) \, \middle| \, \ b(y,\tau) \ = \ 0 \ ; \ \ y \in L^2(\Omega) \, \right\} \ . \tag{3.19}$$

Hence, by general results on saddle point problems (see e. g. §2 in [35]) problem **H** features a unique solution $\{\sigma, u\} \in \mathcal{H}_{\rm div}(\mathcal{M}_h) \times L^2(\Omega)$. Requirement (3.16) remains to be verified. For any $\tau \in H_{\rm div}(\Omega)$ and any $v \in H_0^1(\Omega)$ Green's formula asserts:

$$c(v,\tau) \ \ = \ \ \big(\nabla \cdot \tau \, , \, v \big)_\Omega \ + \ \big(\tau \, , \, \nabla v \big)_\Omega \ \ = \ \ 0 \ .$$

The set of simplicial interfaces in the interior of the domain Ω may be denoted by:

$$\mathcal{E}_h \ := \ \left\{ \, E \subset \Omega \, \middle| \, \Omega_i, \Omega_{i'} \in \mathcal{M}_h \ : \ E = \Omega_i \cap \Omega_{i'} \ \wedge \ \Omega_i \neq \Omega_{i'} \, \right\} \ . \tag{3.20}$$

For $\tau \in \mathcal{H}_{\rm div}(\mathcal{M}_h)$ a function w_τ shall be defined by: $w_\tau|_{\Omega_i} := \nabla \cdot \tau \, |_{\Omega_i} \, \forall \, \Omega_i \in \mathcal{M}_h$. Since $\mathcal{E}_h \subset \Omega$ is a set of measure 0, we conclude: $w_\tau \in L^2(\Omega)$. Under the assumption $c(v,\tau) = 0 \, \forall \, v \in H_0^1(\Omega)$ the weak divergence of τ acting on some test function $v \in C_0^\infty(\Omega)$ can be reformulated as:

$$- \, \big(\tau \, , \, \nabla v \big)_\Omega \ \ = \ \ - \sum_{T \in \mathcal{T}_h} \big(\tau \, , \, \nabla v \big)_T \ \ = \ \ \sum_{T \in \mathcal{T}_h} \big(\nabla \cdot \tau \, , \, v \big)_T \ \ = \ \ \big(w_\tau \, , \, v \big)_\Omega \, .$$

Consequently, the weak divergence of τ can be identified with the square integrable function w_τ. This proves $\tau \in H_{\rm div}(\Omega)$. □

Remark 3.5 When we solve the problem **H** we are generally unable to determine the Lagrange multiplier $w \in H_0^1(\Omega)$ uniquely, as the bilinear form c will not satisfy a compatibility condition in the spirit of (3.18). Let us give an example: If we employ the bilinear form (3.17), we can change a solution $w \in H_0^1(\Omega)$ by any function $\tilde{w} \in C_0^\infty(\Omega)$ whose support is confined to the interior of some element $\Omega_i \in \mathcal{M}_h$.

3.2.2 The Discretisation Procedure

After a regular family $\mathfrak{M} = \{\mathcal{M}_h\}_{h>0}$ of decompositions has been constructed (see e. g. §3.1 in [48]), various trial spaces must be supplied for each decomposition $\mathcal{M}_h \in \mathfrak{M}$ of the domain Ω. Below the set of polynomials of degree less or equal k will be denoted by $\mathbb{P}_k$. These are used to define shape functions of Raviart-Thomas type:

$$\begin{aligned} P^k(\mathcal{M}_h) \ &:= \ \left\{ \, u \in L^2(\Omega) \, \middle| \, u|_{\Omega_i} \in \mathbb{P}_k \ ; \ \Omega_i \in \mathcal{M}_h \, \right\} , \\ B^k(\mathcal{M}_h) \ &:= \ \left\{ \, \tau \in L^2(\Omega, \mathbb{R}^n) \, \middle| \, \tau|_{\Omega_i} \in (\mathbb{P}_k)^n \ ; \ \Omega_i \in \mathcal{M}_h \, \right\} , \\ R^k(\mathcal{M}_h) \ &:= \ \left\{ \, \tau + \vartheta \, x \, \middle| \, \tau \in B^k(\mathcal{M}_h) \, , \, \vartheta \in P^k(\mathcal{M}_h) \, \right\} \ . \end{aligned} \tag{3.21}$$

The degrees of freedom that are associated with these spaces of test functions are well known and need not be recalled here (see e. g. [34, 122] and §2.2 in [48] for details). The canonical interpolation operators are denoted by $\Pi_k : \mathcal{H}_{\rm div}(\mathcal{M}_h) \longrightarrow R_k(\mathcal{M}_h)$ respectively $\hat{\Pi}_k : \mathcal{H}_{\rm div}(\mathcal{M}_h) \longrightarrow B_k(\mathcal{M}_h)$. For the first of these operators the following approximation result is known:

$$\|\!|\, \tau \ - \ \Pi_k \, \tau \,|\!\| \ \ \leq \ \ C \ \ h^{k+1} \ \ \lceil \, \tau \, \rfloor_{k+1} \qquad ; \qquad \tau \in \mathcal{H}_{\rm div}(\mathcal{M}_h) \ . \tag{3.22}$$

Hereby, as in the rest of this paper, the symbol C denotes a generic positive constant, that is independent of the triangulation $\mathcal{M}_h$. For the second operator $\hat{\Pi}_k$ slightly different approximation results have been established:

$$\begin{aligned} \| \tau - \hat{\Pi}_k \tau \| &\leq C\ h^{k+1}\ |\tau|_{k+1} \quad ; \quad \tau \in \mathcal{H}_{\mathrm{div}}(\mathcal{M}_h)\ , \\ \| \tau - \hat{\Pi}_k \tau \| &\leq C\ h^{k}\ \lceil \tau \rfloor_{k+1} \quad ; \quad \tau \in \mathcal{H}_{\mathrm{div}}(\mathcal{M}_h)\ . \end{aligned}$$

Let $P_k : L^2(\Omega) \longrightarrow P^k(\mathcal{M}_h)$ denote the orthogonal projection defined by:

$$\left(P_k v - v\,,\, y_h \right)_\Omega = 0 \quad ; \quad y_h \in P^k(\mathcal{M}_h) \tag{3.23}$$

for any function $v \in L^2(\Omega)$. The interpolation operator Π_k and the projection operator P_k feature the following commutativity property:

$$P_k\,(\nabla\cdot\tau) = \nabla\cdot(\,\Pi_k\,\tau\,) \quad ; \quad \tau \in H_{\mathrm{div}}(\Omega)\ . \tag{3.24}$$

The analogous result for the interpolation operator $\hat{\Pi}_k$ reads:

$$P_{k-1}\,(\nabla\cdot\tau) = \nabla\cdot(\,\hat{\Pi}_k\,\tau\,) \quad ; \quad \tau \in H_{\mathrm{div}}(\Omega)\ . \tag{3.25}$$

The conventional treatment of the problem **H** leads to a discrete formulation $\mathbf{H}_{\mathrm{H}}$, such that the dual variable σ_h is looked for either in $R^k(\mathcal{M}_h)$ or $B^k(\mathcal{M}_h)$. In such a case the approximate solution u_h is contained either in $P^k(\mathcal{M}_h)$ or $P^{k-1}(\mathcal{M}_h)$ and the Lagrange multiplier w_h for the continuity constraints is found in the space $P^k(\mathcal{E}_h)$ (see (3.20) for a definition of the set $\mathcal{E}_h$). Thus a conforming solution $\{\sigma_h, u_h\}$ of the discrete formulation $\mathbf{H}_{\mathrm{H}}$ is obtained: $\sigma_h \in H_{\mathrm{div}}(\Omega)$. Processing of the Lagrange multiplier $w_h \in P^k(\mathcal{E}_h)$ an extension $u_h^* \notin H^1(\Omega)$ can be supplied [9], that provides a more accurate approximation to the actual solution u than u_h does.

Subsequently, the requirement $\sigma_h \in H_{\mathrm{div}}(\Omega)$ will be dispensed with. The finite dimensional trial space $S_h \subset \mathcal{H}_{\mathrm{div}}(\mathcal{M}_h)$, in which the dual variable σ_h is to be found, will be either $R^k(\mathcal{M}_h)$ or $B^k(\mathcal{M}_h)$. The corresponding trial space $U_h \subset L^2(\Omega)$ for the primal variable u_h will read $P^k(\mathcal{M}_h)$ respectively $P^{k-1}(\mathcal{M}_h)$. The Lagrange multiplier w_h for the continuity constraints, however, will be sought in the trial space

$$W_h := P^{k+1}(\mathcal{M}_h) \cap H_0^1(\Omega)\ . \tag{3.26}$$

The degrees of freedom associated with this space are thereby supposed to be of Lagrange type. Thus the following discrete formulation $\mathbf{H}_{\mathrm{H}}$ is arrived at:

Problem $\mathbf{H}_{\mathrm{H}}$ *Given some continuous linear functional $\phi : \mathcal{H}_{\mathrm{div}}(\mathcal{M}_h) \longrightarrow \mathbb{R}$ and some function $f \in L^2(\Omega)$, find a triple $\{\sigma_h, u_h, w_h\} \in S_h \times U_h \times W_h$, such that for any $\{\tau_h, y_h, z_h\} \in S_h \times U_h \times W_h$ the following equations hold:*

$$\left|\begin{array}{lllllll} a(\tau_h,\sigma_h) & + & b(u_h,\tau_h) & - & c(w_h,\tau_h) & = & \phi(\tau_h) \\ b(y_h,\sigma_h) & & & & & = & (f, y_h)_\Omega \\ c(z_h,\sigma_h) & & & & & = & 0 \end{array}\right| \ .$$

Before the proof of convergence can be carried out, it is necessary to assert, that the problem $\mathbf{H}_{\mathrm{H}}$ is well defined for either choice of S_h respectively U_h:

Proposition 3.3 *Assuming the trial spaces $W_h = P^{k+1}(\mathcal{M}_h) \cap H_0^1(\Omega)$, $U_h = P^k(\mathcal{M}_h)$ and $S_h = R^k(\mathcal{M}_h)$ respectively $U_h = P^{k-1}(\mathcal{M}_h)$ and $S_h = B^k(\mathcal{M}_h)$ are employed to discretise problem **H**, its discrete counterpart $\boldsymbol{H}_H$ features an unique solution $\{\sigma_h, u_h\} \in S_h \times U_h$.*

Proof An unique solution $\{\sigma_h, u_h\} \in S_h \times U_h$ of problem $\mathbf{H}_{\mathrm{H}}$ is known to exist if

$$\begin{aligned} B_h &:= \left\{\ \tau_h \in S_h \ \middle|\ b(y_h,\tau_h) = 0\ ;\ y_h \in U_h\ \right\} \subset B\ , \\ B_h^* &:= \left\{\ v_h \in U_h \ \middle|\ b(v_h,\tau_h) = 0\ ;\ \tau_h \in S_h\ \right\} = \{0\} \end{aligned}$$

holds (again see §2 in [35]) with the set B defined by (3.19). The above condition requires the kernel of the discrete divergence operator to consists only of solenoidal vector fields. In addition, the discrete operator as a map from S_h into U_h must be surjective. If we suppose that $\tau_h \in B_h$ is contained in the set $R^k(\mathcal{M}_h)$, the divergence $\nabla\cdot\tau_h$ is an element of the space $P^k(\mathcal{M}_h)$. Let us choose only such test functions $v_h \in P^k(\mathcal{M}_h)$, whose support is some premeditated simplex $\Omega_i \in \mathcal{M}_h$. The definition of the set B_h implies:

$$\big(\, q\,,\, \nabla\cdot\tau_h \,\big)_{\Omega_i} \;=\; 0 \quad ; \quad q \in \mathbb{P}_k \qquad \Rightarrow \qquad \nabla\cdot\tau_h\,\big|_{\Omega_i} \;=\; 0 \;\; .$$

Therefore, $\nabla\cdot\tau_h$ vanishes on the whole of Ω, so $\tau_h \in B$ is established. In case the discretisation $S_h = B^k(\mathcal{M}_h)$ is used, $\nabla\cdot\tau_h \in P^{k-1}(\mathcal{M}_h)$ holds. Hence, the very same argument applies with $q \in \mathbb{P}_k$ replaced by $q \in \mathbb{P}_{k-1}$. Since $P_k\nabla\cdot$ and $P_{k-1}\nabla\cdot$ are surjective maps from $H_{\rm div}(\Omega)$ into U_h, (3.24) respectively (3.25) warrants $B_h^* = \{0\}$. □

Remark 3.6 Again we are unable to determine the Lagrange multiplier $w_h \in W_h$ uniquely, unless a compatibility condition can be imposed on the form c:

$$\gamma_0 > 0 \quad : \quad \sup_{\tau_h\in S_h} \frac{c(z_h,\tau_h)}{|\!|\!|\, \tau_h \,|\!|\!|} \;\geq\; \gamma_0 \,\|\, z_h \,\|_{\Omega,1} \quad ; \quad z_h \in W_h \;\; . \tag{3.27}$$

While establishing such a compatibility condition is impossible in the case of the continuous problem **H**, the above condition may be met in certain cases, if the trial spaces involved in the statement of problem $\mathbf{H}_{\rm H}$ are of low order $k \in \{0,1,2\}$.

3.2.3 Proof of Convergence

The aim of this section is to supply estimates of the rate of convergence for the finite element discretisation specified in the previous section. The proof of convergence is split into four propositions in order not to overburden its exposition. First a best approximation result is derived, that is abstract in the sense, that it depends only on the general structure of the variational problem and not on the actual choice of the trial spaces used in its formulation. Based on this result a bound on the rate of convergence for the dual variable $\sigma_h \in S_h$ is established. The third proposition states the convergence properties of the multiplier $u_h \in U_h$. Finally, it is shown that the assumptions, under which the previous two results have been derived, can be met indeed, if the data of problem **H** has a certain regularity.

Proposition 3.4 *Let $\{\sigma,u,w\} \in H_{\rm div}(\Omega)\times L^2(\Omega)\times H_0^1(\Omega)$ denote a solution of the problem **H** and $\{\sigma_h,u_h,w_h\} \in S_h\times U_h\times W_h$ one of problem $\mathbf{H}_H$. There are two constants $\alpha_1,\alpha_2>0$, such that for any $\{\tilde\sigma_h,\tilde u_h,\tilde w_h\} \in S_h\times U_h\times W_h$:*

$$\begin{aligned}\alpha_1 \,\|\, \sigma_h \,-\, \sigma \,\|_\Omega^2 \;\;\leq\;\; & \alpha_2 \,\|\, \sigma - \tilde\sigma_h \,\|_\Omega^2 \;+\; c(\tilde w_h - w_h, \tilde\sigma_h) \;+\; c(\tilde w_h - w, \sigma_h - \tilde\sigma_h) \\ & +\; b(u-\tilde u_h, \sigma_h - \tilde\sigma_h) \;+\; b(\tilde u_h - u_h, \sigma - \tilde\sigma_h) \;\; .\end{aligned}$$

Proof Due to the regularity of the tensor A there is some constant A_∞, such that:

$$A_\infty > 0 \quad : \quad \sup_{x\in\Omega} \,\|\, A(x)\,\xi \,\| \;\leq\; A_\infty \,\|\, \xi \,\| \quad ; \quad \xi \in \mathbb{R}^n \;\; .$$

Hence, the following estimate can be derived with a view to the requirement (3.15):

$$\begin{aligned} A_0 \,\|\, \sigma_h \,-\, \sigma \,\|_\Omega^2 \;\;\leq\;\; & a(\tilde\sigma_h - \sigma, \sigma_h - \sigma) \;+\; a(\sigma_h - \tilde\sigma_h, \sigma_h - \sigma) \\ \leq\;\; & a(\tilde\sigma_h - \sigma, \sigma_h - \sigma) \;+\; c(w_h - w, \sigma_h - \tilde\sigma_h) \\ -\;\; & b(u_h - u, \sigma_h - \tilde\sigma_h) \\ \leq\;\; & A_\infty \;\|\, \tilde\sigma_h \,-\, \sigma \,\|_\Omega \;\|\, \sigma_h \,-\, \sigma \,\|_\Omega \\ +\;\; & c(w_h - \tilde w_h, \sigma_h - \tilde\sigma_h) \;+\; c(\tilde w_h - w, \sigma_h - \tilde\sigma_h) \\ +\;\; & b(u - \tilde u_h, \sigma_h - \tilde\sigma_h) \;+\; b(\tilde u_h - u_h, \sigma_h - \tilde\sigma_h) \\ \leq\;\; & A_0/2 \,\|\, \sigma_h \,-\, \sigma \,\|_\Omega^2 \;+\; A_\infty^2\, A_0^{-1} \,\|\, \tilde\sigma_h \,-\, \sigma \,\|_\Omega^2 \\ +\;\; & c(\tilde w_h - w_h, \tilde\sigma_h) \;+\; c(\tilde w_h - w, \sigma_h - \tilde\sigma_h) \\ +\;\; & b(u - \tilde u_h, \sigma_h - \tilde\sigma_h) \;+\; b(\tilde u_h - u_h, \sigma - \tilde\sigma_h) \;\; . \end{aligned}$$

To finish the proof the first term on the right hand side must be absorbed into the left hand side of the last inequality. □

Let $Q_k : H_0^1(\Omega) \longrightarrow W_h$ denote the orthogonal projection defined by

$$\left(v \, - \, Q_k \, v \, , \, z_h \right)_\Omega \;\; = \;\; 0 \qquad ; \qquad z_h \; \in \; W_h \;\; .$$

The degrees of freedom associated with the trial space W_h are supposed to be of Lagrange type. Therefore, the following approximation result can be established for any sufficiently regular function $v \in H^{k+1}(\Omega)$:

$$\| \, v \, - \, Q_k \, v \, \|_{\Omega,l} \;\; \leq \;\; C \;\; h^{k-l+1} \;\; | \, v \, |_{\Omega,k+1} \;\; . \tag{3.28}$$

Hereby, $l \in \{0, 1, \ldots k\}$ denotes an arbitrary integer (see e. g. §3.2 in [48]). A similar result holds for the projection operator $P_k : L^2(\Omega) \longrightarrow U_h$ defined by (3.23):

$$\| \, v \, - \, P_k \, v \, \|_{\Omega,l} \;\; \leq \;\; C \;\; h^{k-l+1} \;\; | \, v \, |_{\Omega,k+1}$$

for any $v \in H^{k+1}(\Omega)$. With all the prerequisites in place the rate of convergence for the dual variable σ_h can now be inferred from proposition 3.4:

Proposition 3.5 *Under the assumptions $\sigma \in H^{k+1}(\Omega, \mathbb{R}^n)$ and $w \in H_0^{k+2}(\Omega)$ the finite element scheme described by (3.21) and (3.26) yields an approximate solution σ_h of problem **H** either in $R^k(\mathcal{M}_h)$ or $B^k(\mathcal{M}_h)$, such that:*

$$\| \, \sigma \, - \, \sigma_h \, \|_\Omega \;\; \leq \;\; C \, h^{k+1} \;\; \left\{ \, | \, \sigma \, |_{\Omega,k+1} \, + \, | \, w \, |_{\Omega,k+2} \, \right\} \;\; .$$

Proof Due to (3.23) and (3.24) the choice $\tilde{\sigma}_h := \Pi_k \, \sigma$, $\tilde{u}_h := P_k \, u$ and $\tilde{w}_h := Q_{k+1} \, w$ turns the abstract error bound stated in proposition 3.4 into the following estimate:

$$\begin{aligned} \alpha_1 \, \| \, \sigma_h - \sigma \, \|_\Omega^2 \;\; & \leq \;\; c(\, Q_{k+1} w - w_h \, , \, \Pi_k \, \sigma \,) \;\; + \;\; c(\, Q_{k+1} w - w, \, \sigma_h - \Pi_k \, \sigma \,) \\ & + \;\; \alpha_2 \, \| \, \sigma \, - \, \Pi_k \, \sigma \, \|_\Omega^2 \;\; . \end{aligned} \tag{3.29}$$

If the space $B^k(\mathcal{M}_h)$ is used instead of $R^k(\mathcal{M}_h)$ the same holds true for the choice $\tilde{\sigma}_h := \hat{\Pi}_k \, \sigma$ and $\tilde{u}_h := P_{k-1} \, u$, as $\nabla \cdot \tilde{\sigma}_h$ is contained in $P^{k-1}(\mathcal{M}_h)$ for any $\tilde{\sigma}_h \in B^k(\mathcal{M}_h)$. Since σ is contained in $H_{\mathrm{div}}(\Omega)$ by assumption, so are $\Pi_k \, \sigma$ and $\hat{\Pi}_k \, \sigma$. Furthermore, $Q_{k+1} \, w - w_h \in H_0^1(\Omega)$ holds by construction, so the first expression on the right hand side of (3.29) can be dropped by virtue of (3.16). The second expression on the right hand side of (3.29) can be bounded by:

$$\begin{aligned} c(\, Q_{k+1} w - w, \, \sigma_h - \Pi_k \, \sigma \,) \;\; & = \;\; \sum_{\Omega_i \in \mathcal{M}_h} \int\limits_{\Omega_i} (\, Q_{k+1} w - w \,) \, \nabla \cdot (\, \sigma_h - \Pi_k \sigma \,) \;\; dx \\ & + \;\; \sum_{\Omega_i \in \mathcal{M}_h} \int\limits_{\Omega_i} \nabla (\, Q_{k+1} w - w \,) \, (\, \sigma_h - \Pi_k \sigma \,) \;\; dx \\ & \leq \;\; \sum_{\Omega_i \in \mathcal{M}_h} \int\limits_{\Omega_i} P_k (Q_{k+1} w - w) \, \nabla \cdot \, (\sigma_h - \Pi_k \sigma) \;\; dx \\ & + \;\; \sum_{\Omega_i \in \mathcal{M}_h} \; | \, Q_{k+1} w - w \, |_{\Omega_i,1} \; \| \, \sigma_h - \Pi_k \, \sigma \, \|_{\Omega_i} \\ & \leq \;\; \sum_{\Omega_i \in \mathcal{M}_h} \int\limits_{\Omega_i} P_k (Q_{k+1} w - w) \, \nabla \cdot \, (\sigma_h - \sigma) \;\; dx \\ & + \;\; \frac{1}{\alpha_1} \, | \, Q_{k+1} w - w \, |_{\Omega,1}^2 \;\; + \;\; \frac{\alpha_1}{4} \, \| \, \sigma_h - \Pi_k \, \sigma \, \|^2 \; . \end{aligned}$$

Since $b(z_h, \sigma_h - \sigma) = 0$ holds for any $z_h \in P^k(\mathcal{M}_h)$ the last integral expression vanishes. A similar argument can be used, if the trial space $B^k(\mathcal{M}_h)$ is employed instead of $R^k(\mathcal{M}_h)$. In this

case the projector P_k is to be replaced by P_{k-1}. From the inequality above, from (3.22) and from (3.29) the following estimate can be inferred:

$$\frac{\alpha_1}{2} \| \sigma_h - \sigma \|_\Omega^2 \leq \frac{1}{\alpha_1} | Q_{k+1} w - w |_{\Omega,1}^2 + C^2 \left(\alpha_2 + \frac{\alpha_1}{2} \right) | \sigma |_{\Omega,k+1}^2 h^{2k+2} .$$

Due to the presupposed regularity of the Lagrange multiplier w the approximation result (3.28) can be invoked to finish the proof. □

Since the bilinear form c as specified by (3.17) does not satisfy any compatibility condition in the spirit of (3.18) the Lagrange multiplier $w \in H_0^1(\Omega)$ is not uniquely determined. Hence, the regularity requirements stated in proposition 3.5 have to be understood in the sense, that there has to be at least one solution w that is sufficiently regular. The compatibility condition (3.27) for the finite dimensional trial spaces S_h and W_h may be met in a number of cases. However, there is no general rule how to obtain an unique w_h straight from the formulation of problem $\mathbf{H}_\mathrm{H}$. Unfortunately, this difficulty has an impact on the proof of convergence for the quantity u_h, since the conventional approach based on exploiting the compatibility condition (3.18) fails to decouple the Lagrange multipliers u_h and w_h. By resorting to a duality argument patterned after [58] a proof can be supplied nevertheless.

Proposition 3.6 *If we assume an analytical solution $\{\sigma, u\}$ of problem $\boldsymbol{H}$ to be contained in the space $H_{\mathrm{div}}^{k+1}(\mathcal{M}_h) \times H_0^{k+2}(\Omega)$, we find that the numerical solution $\{\sigma_h, u_h\} \in R^k(\mathcal{M}_h) \times P^k(\mathcal{M}_h)$ respectively $\{\hat{\sigma}_h, \hat{u}_h\} \in B^k(\mathcal{M}_h) \times P^{k-1}(\mathcal{M}_h)$ satisfies the estimate:*

$$\begin{aligned} \| P_k u - u_h \|_\Omega &\leq C\ h^{k+2}\ \left\{ | w |_{\Omega,k+2} + \lceil \sigma \rfloor_{\Omega,k+1} \right\} , \\ \| P_{k-1} u - \hat{u}_h \|_\Omega &\leq C\ h^{k+2-\delta_{1,k}}\ \left\{ | w |_{\Omega,k+2} + \lVert\!\lceil \sigma \rfloor\!\rVert_{\Omega,k+1} \right\} . \end{aligned}$$

Proof Due to (3.15) the differential equation $\nabla\cdot (A^{-T}\nabla z) = P_{k-1}u - \hat{u}_h$ is well defined. Its solution $z \in H_0^1(\Omega)$ is contained in $H^2(\Omega)$ as the domain Ω is convex by assumption and the right hand side is a square integrable function. Hence, there is some constant $C > 0$, such that:

$$\| z \|_{\Omega,2} \leq C\ \| P_{k-1} u - \hat{u}_h \|_\Omega \quad . \tag{3.30}$$

Using the abbreviation $\xi := A^{-T}\nabla z \in H_{\mathrm{div}}(\Omega)$ the following transformations may be carried out with a view to (3.16) and (3.25):

$$\begin{aligned} \| P_{k-1} u - \hat{u}_h \|_\Omega^2 &= (P_{k-1} u - \hat{u}_h , \nabla\cdot \xi)_\Omega = (P_{k-1} u - \hat{u}_h , \nabla\cdot \hat{\Pi}_k \xi)_\Omega \\ &= (u - \hat{u}_h , \nabla\cdot \hat{\Pi}_k \xi)_\Omega = b(u - \hat{u}_h , \hat{\Pi}_k \xi) \\ &= c(w - \hat{w}_h , \hat{\Pi}_k \xi) - a(\hat{\Pi}_k \xi , \sigma - \hat{\sigma}_h) \\ &= a(\xi - \hat{\Pi}_k \xi , \sigma - \hat{\sigma}_h) - a(\xi , \sigma - \hat{\sigma}_h) \quad . \end{aligned}$$

As the requirements of proposition 3.5 are met, the first expression in the equation above can be estimated in the following manner:

$$a(\xi - \hat{\Pi}_k \xi, \sigma - \hat{\sigma}_h) \leq C \| \xi - \hat{\Pi}_k \xi \|_\Omega \| \sigma - \hat{\sigma}_h \|_\Omega \leq C' | z |_{\Omega,2} h^{k+2} .$$

By definition of the dual problem the second expression can be rewritten as:

$$\begin{aligned} a(\xi, \sigma - \hat{\sigma}_h) &= c(z , \sigma - \hat{\sigma}_h) - b(z , \sigma - \hat{\sigma}_h) \\ &= c(z - Q_{k+1} z , \sigma - \hat{\sigma}_h) - b(z - P_{k-1} z , \sigma - \hat{\sigma}_h) \\ &= \big(\nabla(z - Q_{k+1} z) , \sigma - \hat{\sigma}_h\big)_\Omega - b(Q_{k+1} z - P_{k-1} z , \sigma - \hat{\sigma}_h) . \end{aligned}$$

While the first term on the right hand side of this equation can be bounded by invoking (3.28) and proposition 3.5, the second term requires a small detour:

$$\begin{aligned} \| \nabla\cdot \sigma - \nabla\cdot \hat{\sigma}_h \|_\Omega^2 &= b(\nabla\cdot (\sigma - \hat{\sigma}_h) , \sigma - \hat{\sigma}_h) \\ &= b(\nabla\cdot (\sigma - \hat{\sigma}_h) - P_{k-1} \nabla\cdot (\sigma - \hat{\sigma}_h) , \sigma - \hat{\sigma}_h) \\ &\leq C\ h^k\ | \nabla\cdot \sigma |_{\Omega,k}\ \| \nabla\cdot \sigma - \nabla\cdot \hat{\sigma}_h \|_\Omega \quad . \end{aligned}$$

By combining the above results the following estimate is obtained:

$$\| P_{k-1}\, u \,-\, \hat{u}_h \|_\Omega^2 \;\;\leq\;\; C\, h^{k+2}\, |\, z\,|_{\Omega,2} \;+\; C'\, h^k\, \| Q_{k+1} z \,-\, P_{k-1} z \|_\Omega \;\; .$$

In the special case $k = 1$ the norm on the right hand side is bounded by:

$$\| Q_{k+1}\, z \,-\, P_{k-1}\, z \,\|_\Omega \;\;\leq\;\; C\, h^2\, |\, z\,|_{\Omega,2} \;+\; C'\, h\, |\, z\,|_{\Omega,1} \;\; .$$

If trial spaces of higher order are used this estimate can be improved and reads:

$$\| Q_{k+1}\, z \,-\, P_{k-1}\, z \,\|_\Omega \;\;\leq\;\; C\, h^2\, |\, z\,|_{\Omega,2} \;\; .$$

Using these estimates along with (3.30) and tracing the dependence of the various constants on the derivatives of σ and w the second part of the proposition 3.6 is obtained. To demonstrate the first part, the projector P_{k-1} is to be replaced by P_k. The body of the proof carries over with only slight modifications. □

Remark 3.7 Proposition 3.6 does not warrant, that the quantities u_h and $\hat{u}_h$ approximate the analytical solution $u \in L^2(\Omega)$ of problem (3.12) with the rate of convergence $\mathcal{O}(h^{k+2})$. To ensure that the discretisation scheme described in the previous section is indeed convergent with the highest rate of convergence possible, one further condition must be imposed on the regularity of the solution: $u \in H^{k+1}(\Omega)$ if its approximation is sought in $R^k(\mathcal{M}_h) \times P^k(\mathcal{M}_h)$ and $u \in H^k(\Omega)$ otherwise. If this requirement is met, the convergence of the finite element method is a corollary of the above proposition by virtue of the triangle inequality:

$$\| \hat{u}_h \,-\, u \,\|_\Omega \;\;\leq\;\; \| \hat{u}_h \,-\, P_{k-1}\, u \,\|_\Omega \;+\; \| P_{k-1}\, u \,-\, u \,\|_\Omega \;\;=\;\; \mathcal{O}(h^k) \;\; .$$

An upper bound for the error $\| u_h \,-\, u \,\|_{\Omega,2}$ may be obtained in an analogous fashion.

Proposition 3.7 *Under the assumption $A \in W^{k+1,\infty}(\Omega, \mathbb{R}^{n\times n})$ a solution of problem **H** can be found in $H^{k+1}(\Omega, \mathbb{R}^n) \times H^{k+1}(\Omega) \times H_0^{k+2}(\Omega)$, if $f \in H^k(\Omega)$ holds and if there is some vector field $\psi \in H_{div}^{k+1}(\Omega)$ with the property:*

$$\phi(\tau) \;\;=\;\; \sum_{\Omega_i \in \mathcal{M}_h} \Big\{ \big(\, \psi \,,\, \tau \,\big)_{\Omega_i} \;+\; \big(\, \nabla\cdot \psi \,,\, \nabla\cdot \tau \,\big)_{\Omega_i} \Big\} \qquad ; \qquad \tau \;\in\; \mathcal{H}_{div}(\Omega) \;\; .$$

Proof Let $\omega \in H_0^1(\Omega)$ denote the solution of the equation $\nabla\cdot (A^{-1}\nabla\omega) = f$. Since the domain Ω is convex and $f \in H^k(\Omega)$ holds by assumption, the elliptic regularity of this boundary value problem ensures: $\omega \in H^{k+2}(\Omega)$. As there is an element $\eta \in H^{k+2}(\Omega) \cap H_0^1(\Omega)$ with the property $\nabla\cdot (A^{-1}\nabla\eta) = \nabla\cdot (A^{-1}\psi)$, a vector field $\psi_0 \in H^{k+1}(\Omega, \mathbb{R}^n)$ can be determined, such that:

$$\psi \;\;=\;\; A\,\psi_0 \;+\; \nabla\eta \qquad \text{and:} \qquad \nabla\cdot \psi_0 \;=\; 0 \;\; .$$

We will demonstrate, that the vector field $\sigma := \psi_0 + A^{-1}\nabla\omega \in H^{k+1}(\Omega, \mathbb{R}^n)$ and the functions $u := \omega - \eta + \nabla\cdot \psi \in H^{k+1}(\Omega)$ and $w := \omega - \eta \in H_0^{k+2}(\Omega)$ constitute a solution of problem **H**. By construction the second and the third equation hold true for any test function $y \in L^2(\Omega)$ and any $z \in H_0^1(\Omega)$. Let $\tau \in \mathcal{H}_{\rm div}(\Omega)$ designate a test field. The first equation reads:

$$\begin{aligned}
&\sum_{\Omega_i \in \mathcal{M}_h} \Big\{ \big(\, \tau \,,\, A\cdot\sigma \,\big)_{\Omega_i} \;+\; \big(\, u - w \,,\, \nabla\cdot \tau \,\big)_{\Omega_i} \;-\; \big(\, \nabla w \,,\, \tau \,\big)_{\Omega_i} \Big\} \\
=\;\; &\sum_{\Omega_i \in \mathcal{M}_h} \Big\{ \big(\, \tau \,,\, A\cdot\psi_0 + \nabla\omega \,\big)_{\Omega_i} \;+\; \big(\, \nabla\cdot \psi \,,\, \nabla\cdot \tau \,\big)_{\Omega_i} \;-\; \big(\, \nabla\omega - \nabla\eta \,,\, \tau \,\big)_{\Omega_i} \Big\} \\
=\;\; &\sum_{\Omega_i \in \mathcal{M}_h} \Big\{ \big(\, \tau \,,\, A\cdot\psi_0 + \nabla\eta \,\big)_{\Omega_i} \;+\; \big(\, \nabla\cdot \psi \,,\, \nabla\cdot \tau \,\big)_{\Omega_i} \Big\} \;\;=\;\; \phi(\tau) \;\; . \qquad \square
\end{aligned}$$

3.2.4 Processing the Numerical Solution

The Helmholtz splitting on which the proof of proposition 3.7 is based can also be used to demonstrate, that the solution $\{\sigma, u\}$ of problem **H** is under certain conditions on the linear functional ϕ related the solution of the elliptic boundary value problem (3.12). A conventional analysis of the dual formulation leads to the choice

$$\phi(\tau) \quad := \quad \oint\limits_{\partial\Omega} u_0 \, \tau_n \; ds \qquad ; \qquad \tau \in H_{\mathrm{div}}(\Omega) \;\; , \tag{3.31}$$

whereby $u_0 \in H^1(\Omega)$ describes the inhomogeneous Dirichlet boundary data. Such an analysis is based on the assumption, that all the vector fields involved in the calculus are at least contained in $H_{\mathrm{div}}(\Omega)$. However, this assumption can no longer be maintained if the nonconforming discretisation scheme is employed we have introduced in subsection 3.2.2. Hence, it is not possible to keep (3.31) and simply consider the extension $\phi^* : \mathcal{H}_{\mathrm{div}}(\mathcal{M}_h) \longrightarrow \mathbb{R}$, that is defined by the same boundary integral. If

$$\phi^*(\tau) \;\; := \sum_{\Omega_i \in \mathcal{M}_h} \Big\{ \big(u_0 \, , \, \nabla\cdot \tau \big)_{\Omega_i} \, + \, \big(\nabla u_0 \, , \, \tau \big)_{\Omega_i} \Big\} \qquad ; \qquad \tau \in \mathcal{H}_{\mathrm{div}}(\mathcal{M}_h) \tag{3.32}$$

is used instead of (3.31) in the right hand side of problem **H**, the proper solution of (3.12) is obtained, as the proof of proposition 3.7 demonstrates. This proof also shows, that there is a special solution $\{\sigma, u, w\}$ with the property: $u = w + u_0$. Consequently, it seems reasonable to expect, that an improved approximation to the solution u of problem **H** may be obtained from a solution $\{\sigma_h, u_h, w_h\}$ of problem $\mathbf{H}_{\mathrm{H}}$ by post-processing the Lagrange multiplier w_h. Let $\pi_k : H^1_0(\Omega) \longrightarrow P^k(\mathcal{E}_h)$ denote the orthogonal projection defined for any $v \in H^1_0(\Omega)$ by:

$$\sum_{E \in \mathcal{E}_h} \int\limits_E (\, \pi_k v - v \,) \, z_h \; ds \;\; = \;\; 0 \qquad ; \qquad z_h \in P^k(\mathcal{E}_h) \;\; .$$

Proposition 3.8 *Let $\{\sigma_h, u_h, w_h\} \in R^k(\mathcal{M}_h) \times P^k(\mathcal{M}_h) \times W_h$ designate a solution of problem $\boldsymbol{H}_H$ (respectively $\{\hat\sigma_h, \hat u_h, \hat w_h\} \in B^k(\mathcal{M}_h) \times P^{k-1}(\mathcal{M}_h) \times W_h$) and $\{\sigma, u, w\}$ one of problem $\boldsymbol{H}$. Defining $\Sigma := \bigcup_{E \in \mathcal{E}_h} E$ the following estimates hold:*

$$C \, \sqrt{h} \;\; \| \, \pi_k \, (\, w \, - \, w_h \,) \, \|_{\Sigma} \;\; \leq \;\; h \, \| \, \sigma \, - \, \sigma_h \, \|_{\Omega} \, + \, \| \, P_k \, u \, - \, u_h \, \|_{\Omega} \;\; ,$$

$$C \, \sqrt{h} \;\; \| \, \pi_k \, (\, w \, - \, \hat w_h \,) \, \|_{\Sigma} \;\; \leq \;\; h \, \| \, \sigma \, - \, \hat\sigma_h \, \|_{\Omega} \, + \, \| \, P_{k-1} \, u \, - \, \hat u_h \, \|_{\Omega} \;\; .$$

Proof Let $E \in \mathcal{E}_h$ designate an arbitrary interface and choose an element $\Omega_i \in \mathcal{M}_h$ with the property $E \subset \Omega_i$. If the support of some field $\tau_h \in S_h$ is limited to the patch Ω_i, the equation

$$\big(\tau_h \, , \, A \cdot (\sigma - \sigma_h) \big)_{\Omega_i} \;\; + \;\; \big(u - u_h \, , \, \nabla\cdot \tau_h \big)_{\Omega_i} \;\; = \;\; \oint\limits_{\partial\Omega_i} (w - w_h) \, (\tau_h)_n \; ds \tag{3.33}$$

is obtained by subtracting the first equation in the statement of problem $\mathbf{H}_{\mathrm{H}}$ from the first equation in the statement of problem **H**. By definition of the trial space $R^k(\mathcal{M}_h)$ the vector field τ_h may be chosen in such a way, that

$$\oint\limits_{E'} q(s) \, (\tau_h)_n \; ds \quad = \quad \delta_{E,E'} \; \oint\limits_{E} \pi_k(w - w_h) \, q(s) \; ds$$

holds for any polynomial $q \in \mathbb{P}_k$ and any edge $E' \in \mathcal{E}_h$. Further degrees of freedom the vector field τ_h may possess are supposed to be zero. Under this assumption we may exploit the fact, that all norms on a finite dimensional space are equivalent, and thence infer by a scaling argument:

$$\| \, \tau_h \, \|_{\Omega_i} \;\; \leq \;\; C \, \sqrt{h} \;\; \| \, (\tau_h)_n \, \|_{E} \;\; = \;\; C \, \sqrt{h} \;\; \| \, \pi_k \, (w - w_h) \, \|_{E} \;\; .$$

As $\nabla\cdot\tau_h \in P^k(\mathcal{M}_h)$ holds, an inverse inequality can be used to bound the divergence. In addition, the solution u can be replaced by $P_k u$. Hence, equation (3.33) implies:

$$\begin{aligned} \| \pi_k(w - w_h) \|_E^2 &\leq C \, \| \sigma - \sigma_h \|_T \, \| \tau_h \|_{\Omega_i} + \| P_k u - u_h \|_T \, \| \nabla\cdot \tau_h \|_{\Omega_i} \\ &\leq C \sqrt{h} \left\{ \| \sigma - \sigma_h \|_{\Omega_i} + h^{-1} \| P_k u - u_h \|_{\Omega_i} \right\} \| (\tau_h)_n \|_E \, . \end{aligned}$$

After summation of this inequality with respect to $E \in \mathcal{E}_h$ the proof is finished, for no element $\Omega_i \in \mathcal{M}_h$ can appear more than two times on the right hand side of the resulting inequality. If we consider approximation $\{\hat\sigma_h, \hat u_h, \hat w_h\}$ instead of $\{\sigma_h, u_h, w_h\}$ we must replace P_k by P_{k-1}, for $\nabla\cdot\tau_h \in P^{k-1}(\mathcal{M}_h)$ holds in this case. No further changes to the proof are necessary. □

Apart from those degrees of freedom located, so to speak, on its interfaces, a Lagrange element of order $k > n$ features degrees of freedom "inside" the cell, that correspond to shape functions whose support is confined within the element. Using barycentric coordinates $\{\lambda_0, \ldots \lambda_d\}$ any linear combination $q_{k+1} \in \mathbb{P}_{k+1}$ of these shape functions can be written as: $q_{k+1} = \lambda_0 \ldots \lambda_d \, q_{k-n}$ with some $q_{k-n} \in \mathbb{P}_{k-n}$. As the product $\lambda_0 \ldots \lambda_d$ is strictly positive inside the cell, $P_{k-n}\, q_{k+1} = 0$ implies $q_{k+1} = 0$. Hence, it is possible to uniquely define the conforming approximation $u_h^* \in W_h$ of the solution u of (3.12) by requiring $P_{k-n}\, u_h^* = P_{k-n}\, u_h$ along with:

$$u_h^* \big|_E = (w_h + I_{k+1}\, u_0) \big|_E \quad ; \quad E \in \mathcal{E}_h \, . \tag{3.34}$$

Hereby, $I_{k+1}\!:\! H^1(\Omega) \longrightarrow W_h$ denotes the canonical finite element interpolation. In the case $k < n$ the first condition is void. A linear map $\hat Q_{k+1} : H_0^1(\Omega) \longrightarrow P^{k+1}(\mathcal{M}_h) \cap H_0^1(\Omega)$ is specified by:

$$P_{k-n} \left(v - \hat Q_{k+1}\, v \right) = 0 \quad \text{and:} \quad \hat Q_{k+1} v \big|_E = \pi_{k+1}\, v \big|_E \quad ; \quad E \in \mathcal{E}_h$$

for any $v \in H_0^1(\Omega)$. This mapping is continuous and leaves elements from the space W_h invariant. From (3.28) the following approximation property can be inferred:

$$\| u - \hat Q_{k+1} u \|_\Omega = \| u - Q_{k+1}\, u - \hat Q_{k+1}(u - Q_{k+1}\, u) \|_\Omega \leq C\, h^{k+2} \, | u |_{\Omega, k+2} \, .$$

Under the assumption that a polynomial $q \in \mathbb{P}_{k+1}$ attains the value 0 along at least one edge of the interface E, the condition $\pi_k\, q = 0$ implies $q\,|_E = 0$. Starting with those elements adjoining the boundary $\partial\Omega$ it can therefore be demonstrated inductively, that the seminorm

$$|\!|\!| v_h |\!|\!|_k^2 := h \sum_{E \in \mathcal{E}_h} \| \pi_k\, v_h \|_E^2 + \| P_{k-n}\, v_h \|_\Omega^2 \quad ; \quad v_h \in W_h$$

is actually a norm on the space W_h. In the case $k < n$ the second expression on the right side is to be dropped. Since W_h is a space of finite dimension this new norm must be equivalent to the L^2-norm. Hence, there is a constant $K_h := K(\mathcal{M}_h)$, possibly dependent on the simplicial decomposition $\mathcal{M}_h \in \mathfrak{M}$, such that:

$$\| v_h \|_\Omega \leq K_h \, |\!|\!| v_h |\!|\!|_k \quad ; \quad v_h \in W_h \, . \tag{3.35}$$

Proposition 3.9 *If the conditions stated in proposition 3.7 are met, if $u_0 \in H^{k+2}(\Omega)$ holds and if the constant K_h in (3.35) is uniformly bounded, the approximations $u_h^* \in W_h$ and $\hat u_h^* \in W_h$ as defined by $P_{k-n}\, u_h^* = P_{k-n}\, u_h$ and (3.34) satisfy:*

$$\begin{aligned} C \, \| u - u_h^* \|_\Omega &\leq h^{k+2} \left\{ | u |_{\Omega,k+2} + | u_0 |_{\Omega,k+2} + \lceil \sigma \rfloor_{\Omega,k+1} \right\} \, , \\ C \, \| u - \hat u_h^* \|_\Omega &\leq h^{k+2-\delta_{k,1}} \left\{ | u |_{\Omega,k+2} + | u_0 |_{\Omega,k+2} + \lVert\!\lceil \sigma \rfloor\!\rVert_{\Omega,k+1} \right\} . \end{aligned}$$

Proof Since the map $\hat Q_{k+1}$ leaves elements from the space W_h invariant, a scaling argument in the spirit of [49] warrants the existence of a constant C, such that

$$h \sum_{E \in \mathcal{E}_i} \| \pi_{k+1}\, v_h \|_E^2 + \| P_{k-n}\, v_h \|_{\Omega_i}^2 \leq C \, \| v_h \|_{\Omega_i}^2 \quad ; \quad v_h \in P^{k+1}(\mathcal{M}_h)$$

holds uniformly in $\Omega_i \in \mathcal{M}_h$. Thereby, the abbreviation $\mathcal{E}_i := \{ E \in \mathcal{E}_h \,|\, E \subset \partial\Omega_i \}$ has been used. A global estimate is derived by taking the sum with respect to $\Omega_i \in \mathcal{M}_h$:

$$h \sum_{E \in \mathcal{E}_h} \| \pi_{k+1} v_h \|_E^2 \quad \leq \quad C \, \| v_h \|_\Omega^2 \qquad ; \qquad v_h \in P^{k+1}(\mathcal{M}_h) \ . \tag{3.36}$$

Employing the map $\hat{Q}_{k+1}$ and exploiting (3.35) the following bound can be found:

$$\begin{aligned} \| u - u_h^* \|_\Omega \quad &\leq \quad \| u - \hat{Q}_{k+1} u \|_\Omega \ + \ \| \hat{Q}_{k+1} u - u_h^* \|_\Omega \\ &\leq \quad C \ h^{k+2} \, | u |_{\Omega,k+2} \ + \ K_h \ |\!|\!| \hat{Q}_{k+1} u - u_h^* |\!|\!|_k \ . \end{aligned}$$

With a view to the definition of $\hat{Q}_{k+1}$ and due to $P_{k-n} u_h^* = P_{k-n} u_h$ the second part of the mesh dependent norm $|\!|\!| \hat{Q}_{k+1} u - u_h^* |\!|\!|_k$ can be estimated by:

$$\| P_{k-n} (\hat{Q}_{k+1} u - u_h^*) \|_\Omega \ = \ \| P_{k-n} (u - u_h) \|_\Omega \ \leq \ \| P_k u - u_h \|_\Omega$$

thanks to $P_{k-n} u = P_{k-n} P_k u$. In the case $k < n$ this estimate is to be ignored. To bound the other part of the norm proposition 3.8 must be involved. As we have remarked previously, there is a solution $\{\sigma, u, w\}$ of problem **H** with the property: $u = w + u_0$. Using this very solution the following inequality results from (3.34) for any $E \in \mathcal{E}_h$:

$$\begin{aligned} \| \pi_k (\hat{Q}_{k+1} u - u_h^*) \|_E^2 \quad &= \quad \| \pi_k (\pi_{k+1} u - w_h - I_{k+1} u_0) \|_E^2 \\ &\leq \quad 2 \left\{ \| \pi_k (w - w_h) \|_E^2 \ + \ \| \pi_{k+1} u_0 - I_{k+1} u_0 \|_E^2 \right\} \ . \end{aligned}$$

Due to (3.36) summation of the above inequality with respect to $E \in \mathcal{E}_h$ yields:

$$C \, |\!|\!| \hat{Q}_{k+1} u - u_h^* |\!|\!|_k \ \leq \ \| \hat{Q}_{k+1} u_0 - I_{k+1} u_0 \|_\Omega \ + \ \| P_k u - u_h \|_\Omega \ + \ h \, \| \sigma - \sigma_h \|_\Omega \ .$$

Thanks to proposition 3.5 and 3.6 the proof is finished herewith. If the approximation $\hat{u}_h^* \in W_h$ is constructed from the solution $\{\hat{\sigma}_h, \hat{u}_h, \hat{w}_h\} \in B^k(\mathcal{M}_h) \times P^{k-1}(\mathcal{M}_h) \times W_h$ the reasoning does not change. Merely $P_k u$ has to be replaced by $P_{k-1} u$. □

The proposition 3.9 asserts, that an approximation to the solution of the primal problem (3.12) may be found by post processing the Lagrange multipliers for the continuity constraints, that exhibits a higher asymptotic rate of convergence than the Lagrange multipliers for the admissibility constraints do themselves. However, as a prerequisite for such a manipulation of the numerical data to be successful the stability constant in (3.35) must be ascertained to be uniformly bounded in $\mathcal{M}_h \in \mathfrak{M}$.

Proposition 3.10 *If $n = 2$ holds and $k \in \mathbb{N}_0$ is an even integer, the stability constant K_h in estimate (3.35) is uniformly bounded in $\mathcal{M}_h \in \mathfrak{M}$.*

Proof Let $\Omega_i \in \mathcal{M}_h$ denote an arbitrary element. The set of its interfaces may be abbreviated by $\mathcal{E}_i := \{ E \in \mathcal{E}_h \,|\, E \subset \partial\Omega_i \}$. Assuming $q \in \mathbb{P}_{k+1}$ has the property:

$$N_i(q)^2 \quad := \quad h \sum_{E \in \mathcal{E}_i} \| \pi_k q \|_E^2 \ + \ \| P_{k-n} q \|_{\Omega_i}^2 \quad = \quad 0$$

it shall be demonstrated that $q = 0$ holds. The condition $\pi_k q = 0$ implies, that q is a multiple of the Legendre polynomial $P_{k+1}(s)$, if considered as a function of the arc length s along some interface $E \in \mathcal{E}_h$. As such q must attain opposite values at the endpoints of each interface (see e. g. §22 in [1]). Since this is not possible for an odd number of interfaces, q can have no trace on $\partial\Omega_i$. In the case $k < n$ this would imply $q = 0$. If $k \geq n$ holds, the requirement $P_{k-n} q = 0$ enforces $q = 0$. Hence, the functional N_i is definite and can be used as a norm for the space $\mathbb{P}_{k+1}$. A scaling argument in conjunction with the regularity of $\mathcal{M}_h$ asserts that $N_i(q) \leq C \, \| q \|_{\Omega_i}$ holds for any $q \in \mathbb{P}_{k+1}$. Summation over $\Omega_i \in \mathcal{M}_h$ now proves the stability result (3.35) for any $v_h \in P^{k+1}(\mathcal{M}_h)$ with $K_h \leq C$. □

Remark 3.8 Though the proof of proposition 3.10 can neither be applied in the case $n = 3$ nor in the case of k odd, the approximations $u_h^* \in W_h$ and $\hat{u}_h^* \in W_h$ are not meaningless. If k is odd and $n = 2$ for instance, an estimate of the form

$$\sup_{x\in\Omega} |\,(I - P_{k-d})\,v_h(x)\,| \;\leq\; \frac{C}{h^3} \sum_{E\in\mathcal{F}_h} \|\,\pi_k\,v_h\,\|_E^2 \qquad ; \qquad v_h \in W_h$$

can be invoked to demonstrate, that the constant K_h is bounded on reasonable meshes by $C\,h^{-1}$ at the worst. Thereby $\mathcal{F}_h \subset \mathcal{E}_h$ denotes the set of those edges, that form the shortest path from the boundary $\partial\Omega$ to the very simplex, within which the supremum of $(I - P_{k-n})\,v_h$ is attained. In consequence, the function $\hat{u}_h^*$ is still a valid approximation of the solution u to problem (3.12). Furthermore, the asymptotic rate of convergence will still read $\mathcal{O}(h^{k+1})$ and hence will be better than that of $\hat{u}_h$.

Remark 3.9 The case $n = 3$ has not been closely examined by the author. The very simplest discretisation scheme, that can be imagined within the framework of this paper, is based on continuous, piecewise linear shape functions to span the trial space W_h. However, for such linear shape functions the barycentres of the element interfaces are well known to form an unisolvent set (see e. g. [54]). By a scaling argument similar to the one used above $K_h \leq C$ follows immediately.

Remark 3.10 The lowest order case $k = 0$ is special in so far, as an equivalence between the nonconforming hybrid discretisation scheme introduced in section 3.2.2 and a conventional conforming discretisation method for the Dirichlet problem (3.12) can be established under certain assumptions on the tensor A. If $A \in L^\infty(\Omega, \mathbb{R}^{n\times n})$ is constant on each simplex $\Omega_i \in \mathcal{M}_h$ the field $\tau_h := A^{-T}\nabla z_h$ is an admissible test field for any $z_h \in W_h = P^1(\mathcal{M}_h) \cap W_0^1(\Omega)$. With a view to (3.17) and (3.32) the following equation can be inferred from the statement of problem $\mathbf{H}_{\mathrm{H}}$:

$$\sum_{\Omega_i\in\mathcal{M}_h} \big(\, A^{-T}\nabla z_h\,,\, A\,\sigma_h\,\big)_{\Omega_i} \;=\; \sum_{\Omega_i\in\mathcal{M}_h} \big(\,\nabla u_0 + \nabla w_h\,,\, A^{-T}\nabla z_h\,\big)_{\Omega_i} \qquad ; \qquad z_h \in W_h\;.$$

Due to $c(z_h, \sigma_h) = 0$ and $\nabla\cdot\,\sigma_h \in P^0(\mathcal{M}_h)$ its left hand side can be rewritten as:

$$\big(\,\nabla z_h\,,\,\sigma_h\,\big)_\Omega \;=\; -\sum_{\Omega_i\in\mathcal{M}_h} \big(\,z_h\,,\,P_0\,\nabla\cdot\,\sigma_h\,\big)_{\Omega_i} \;=\; -\,b(\,P_0\,z_h, \sigma_h\,) \;=\; -\,\big(\,f,\,P_0\,z_h\,\big)_\Omega\;.$$

By combining these two results a conforming finite element scheme for the primal formulation of the inhomogeneous Dirichlet problem is eventually discovered:

$$w_h \in W_h \qquad : \qquad \big(\,\nabla z_h, A^{-1}\nabla(u_0 + w_h)\,\big)_\Omega \;=\; -\,\big(\,P_0\,f, z_h\,\big)_\Omega \qquad ; \qquad z_h \in W_h\;.$$

3.3 Hypercycle Estimates in a Finite Element Context

The analysis of the two applications we have presented in chapter 2 aimed at establishing efficiency results for duality based a posteriori error majorants, that were of a somewhat academic nature: how close could we possibly get to the true approximation error and what would the minimiser of the error majorant look like. For all practical purposes, however, finding the minimiser of an error majorant is at least as difficult as solving the primal formulation. Hence, we have to accept the deteriorating impact of numerical approximations on the efficiency of our error estimates. In the following paragraphs we will present a number of readily computable estimates, which can be identified as special instances of the abstract theory we have developed throughout chapter 1.

3.3.1 A posteriori estimates for the Laplace Problem

Using the same notation we have employed throughout section 2.1.3 let us consider the Dirichlet problem with homogeneous boundary conditions. For simplicity, we will assume, that $\Omega \subset \mathbb{R}^n$ is a convex domain with a polygonal boundary. Furthermore, we will suppose, that a simplicial decomposition $\mathcal{M}_h$ of the domain Ω has been constructed, with the help of which a finite element

approximation $x_h \in H^1_0(\Omega)$ to the exact solution $x_0 \in H^1_0(\Omega)$ has been produced. Below we need to refer to a number of function spaces, whose definition let us recall for our convenience:

$$\begin{aligned} P^k(\mathcal{M}_h) &:= \left\{ u \in L^2(\Omega) \,\middle|\, u|_{\Omega_i} \in \mathbb{P}_k \;;\; \Omega_i \in \mathcal{M}_h \right\}, \\ B^k(\mathcal{M}_h) &:= \left\{ \tau \in L^2(\Omega, \mathbb{R}^n) \,\middle|\, \tau|_{\Omega_i} \in (\mathbb{P}_k)^n \;;\; \Omega_i \in \mathcal{M}_h \right\}, \\ R^k(\mathcal{M}_h) &:= \left\{ \tau + \vartheta\, x \,\middle|\, \tau \in B^k(\mathcal{M}_h)\,,\; \vartheta \in P^k(\mathcal{M}_h) \right\} . \end{aligned}$$

Hereby, the symbol $\mathbb{P}_k$ shall denote the set of all polynomials in n variables with a degree less or equal $k \in \mathbb{N}_0$. We extend the above notation, which we have used throughout section 3.2.2, by introducing subspaces of functions respectively vector fields with a somewhat higher regularity:

$$P^k_c(\mathcal{M}_h) := P^k(\mathcal{M}_h) \cap H^1(\Omega) \qquad ; \qquad B^k_c(\mathcal{M}_h) := B^k(\mathcal{M}_h) \cap H_{\mathrm{div}}(\Omega) \;.$$

The Raviart-Thomas space $R^k_c(\mathcal{M}_h)$ is defined in an analogous fashion. The degrees of freedom associated with these spaces are well known and need not be discussed. (Detailed information can be found e. g. in [34, 122] or in §2.2 of [48].) In the following let us assume, that the numerical approximation x_h be contained in the space $P^k_0(\mathcal{M}_h) := P^k(\mathcal{M}_h) \cap H^1_0(\Omega)$ for some $k \geq 1$ and has been obtained by solving the finite dimensional variational formulation:

$$\langle\, \nabla x_h, \nabla v_h \,\rangle_\Omega \quad = \quad (\, f, v_h \,)_\Omega \qquad ; \qquad v_h \in P^k_0(\mathcal{M}_h) \;. \tag{3.37}$$

If the data of the Dirichlet problem is sufficiently smooth to warrant its analytical solution being contained in the space $H^{k+1}(\Omega) \cap H^1_0(\Omega)$, we can obtain from the Lax-Milgram theorem [99] the following a priori bound for the approximation error:

$$|\, x_h - x_0 \,|_{\Omega,1} \quad \leq \quad C(\Omega)\, h^k \;|\, x_0 \,|_{\Omega,k+1} \quad \leq \quad C'(\Omega)\, h^k \;|\, f \,|_{\Omega,k-1} \;. \tag{3.38}$$

Against this estimate we have to compare the performance of any a posteriori error estimator, which we can derive either from (2.7) or from (2.8) by choosing a suitable vector field σ_h from one of the finite dimensional spaces just introduced.

A conventional a posteriori error estimate

Though the numerical solution x_h is supposed to be conforming, its gradient $\nabla x_h \in P^{k-1}(\mathcal{M}_h)$ is most likely not a continuous vector field. Hence, we may not assume Δx_h to be square integrable. Choosing the field $\sigma_h := \nabla x_h$ is possible nevertheless, if we want to evaluate the sharper of the two error bounds (2.7) and (2.8). In taking the limit $\kappa \to 0$ we recover the well known a priori estimate for the approximation error:

$$|\, x_h \,-\, x_0 \,|^2_{\Omega,1} \quad \leq \quad |\, \Delta x_h \,+\, f \,|^2_{\Omega,-1} \;. \tag{3.39}$$

We are unable to evaluate the above error bound, unless we require the numerical approximation to possess the best approximation property implied by (3.37). If that condition is met and the function f is at least square integrable, we may invoke standard results (see e. g. §3.2 in [48]) on the approximation properties of the canonical finite element interpolation $I_k : C(\Omega) \longrightarrow P^k_c(\Omega)$ in order to bound the dual norm of the residual. Let E denote an interior edge which is shared by the elements M_1 and M_2. We define the jump $[\sigma]$ across E of a sufficiently regular vector field σ with the help of the outward pointing normal vectors $n_1 \in \mathbb{R}^2$ and $n_2 = -n_1$ perpendicular to E:

$$[\sigma](\xi) \quad := \quad n_1 \cdot \sigma\,\big|_{M_1}(\xi) \;+\; n_2 \cdot \sigma\,\big|_{M_2}(\xi) \qquad ; \qquad \xi \in E \;.$$

Let us suppose, that the mesh $\mathcal{M}_h$ is regular in the sense of definition 3.2. We abbreviate by $\mathcal{E}_h$ the set of all edges, which are interior to the mesh $\mathcal{M}_h$. By choosing $v_h = I_k\, v$ as a test function in (3.37) and performing a partial integration on each patch $M \in \mathcal{M}_h$ we find:

$$|\, x_h - x_0 \,|_{\Omega,1} \quad \leq \quad \sup_{v \in H^1_0(\Omega)} \frac{1}{|\, v \,|_{\Omega,1}} \sum_{M \in \mathcal{M}_h} \left\{ \int_M (f + \Delta x_h)(v - v_h) \;-\; \frac{1}{2} \oint_{\partial M} (v - v_h) \left[\frac{\partial x_h}{\partial n} \right] \right\}$$

$$\leq \sup_{v \in H_0^1(\Omega)} \frac{1}{|\, v \,|_{\Omega,1}} \left\{ \sum_{M \in \mathcal{M}_h} C_M \, h_M \, |\, v \,|_{M,1} \; \| \, f + \Delta x_h \, \|_M \quad + \right.$$

$$\left. + \quad \sum_{E \in \mathcal{E}_h} \hat{C}_M \, \sqrt{h_E} \; |\, v \,|_{M,1} \; \left\| \left[\frac{\partial x_h}{\partial n} \right] \right\|_E \right\}$$

$$\leq \quad C \quad \left\{ \sum_{M \in \mathcal{M}_h} h_M^2 \, \| \, f + \Delta x_h \, \|_M^2 \; + \; \sum_{E \in \mathcal{E}_h} h_E \left\| \left[\frac{\partial x_h}{\partial n} \right] \right\|_E^2 \right\}^{1/2} \; .$$

The above inequality renders just the usual explicit energy error estimator, which is based on the evaluation of element residuals and jump terms for the numerical fluxes (see e. g. [5, 138]).

Smoothing the primal approximation

Since the computation of the edge contributions may require substantial technical effort, it seems reasonable to employ vector fields σ_h of a somewhat higher regularity. If the divergence of such fields is square integrable, the residual contributions to the error majorant (2.8) can be evaluated immediately: in such a case we can employ the same numerical technology for providing error bounds as for assembling the finite element matrices. Our simplest choice for the dual parameter is a product ansatz of the form $\sigma_h \in (P_c^k(\Omega))^n$, as it does not force us to manage different types of shape functions on the same mesh. A projection $\Pi : L^2(\Omega, \mathbb{R}^n) \longrightarrow (P_c^k(\mathcal{M}))^n$ may be defined in several ways. Each degree of freedom, the space $P_c^k(\mathcal{M})$ is equipped with, can be identified with some point $\xi_j \in \Omega$. The associated shape function let us denote by $\psi_j \in P_c^k(\mathcal{M})$. We may specify an interpolation operator in the spirit of Clément [52]:

$$\Pi \, \sigma \quad := \quad \sum_{M \in \mathcal{M}_h} |M| \sum_{\xi_j \in \operatorname{clo} M} \frac{\sigma|_M(\xi_j)}{|\operatorname{supp} \psi_j|} \; \psi_j \quad ,$$

provided the vector field σ is continuous on each element. Since the vector field ∇x_h meets this requirement, $\Pi(\nabla x_h)$ is a viable choice for the dual parameter in (2.8). The main advantage of such a procedure lies in the fact, that the action of the interpolation operator Π can be computed easily by a single sweep across the mesh. The disadvantage of such a choice lies in the problem of asserting upper bounds in terms of the mesh parameter h for the two residual contributions, which constitute the error majorant (2.8). In [94] a similar averaging method is discussed for piecewise linear shape functions on highly structured meshes: The resulting field is found to approximate the gradient of an analytical solution $x_0 \in H^3(\Omega)$ with the order $\mathcal{O}(h^2)$. Since the averaging methods differ near the boundary of the computational domain, a closer look at the proofs in [94] indicate, that the operator Π as defined above will generate a sort of boundary layer. Results pertaining also to unstructured meshes have been published in [83]: However, these fail to answer the important question, how the second part $\tilde{K}^*$ of the error estimate (2.8) is affected by the averaging procedure. Perhaps even less amenable to an analysis is the construction of a smoothed approximation $\hat{\sigma}_h \in H_{\text{div}}(\Omega)$ to the gradient ∇x_h with the help of a L^2-projection:

$$\hat{\sigma}_h \in (P_c^k(\mathcal{M}))^n \quad : \quad \langle \, \hat{\sigma}_h - \nabla x_h, \tau_h \, \rangle_\Omega \; = \; 0 \quad ; \quad \tau_h \in (P_c^k(\mathcal{M}))^n \quad .$$

As the condition number of the mass matrix is independent of the mesh parameter h, the above system can be solved with optimal numerical complexity using a CGS iteration [133] for instance. Assuming the analytical solution x_0 to be contained at least in the space $H^{k+2}(\Omega)$ the first part $\hat{M}_F(x_h, \hat{\sigma}_h)$ of the hypercycle estimate (2.8) can be bounded by:

$$\| \, \nabla x_h - \hat{\sigma}_h \, \|_\Omega \quad \leq \quad |\, x_h - x_0 \,|_{\Omega,1} \; + \; \| \, \nabla x_0 - I_k \nabla x_0 \, \|_\Omega \quad = \quad |\, x_h - x_0 \,|_{\Omega,1} \; + \; \mathcal{O}(h^{k+1}) \; .$$

The second part $\tilde{K}^*$ of the error bound (2.8) proves to be the difficult one. At present, we can not offer any satisfactory analysis of the residual $\| \nabla \cdot \hat{\sigma}_h + f \|_\Omega$ in terms of the mesh parameter h. In any event, we must not presume this expression to be a higher order perturbation to the first part of the error majorant. Once the field σ_h is fixed, we can determine the optimal equilibration parameter κ^* by minimising the right hand side of (2.8) with respect to $\kappa > 0$. We find:

$$\kappa^* \quad := \quad \lambda_0 \; \frac{\| \, \nabla x_h - \sigma_h \, \|_\Omega}{\| \, \nabla \cdot \sigma_h + f \, \|_\Omega} \quad .$$

Consequently, we may improve the estimate (2.8) using $\kappa = \kappa^*$ and taking the square root:

$$| x_h - x_0 |_{\Omega,1} \quad \leq \quad M_0(\sigma_h) \quad := \quad \| \nabla x_h \, - \, \sigma_h \|_\Omega \; + \; \lambda_0^{-1} \, \| \nabla \cdot \sigma_h \, + \, f \|_\Omega \quad .$$

The error bound M_0 thus defined is valid for any vector field $\sigma_h \in H_{\rm div}(\Omega)$. However, neither of the two choices $\Pi(\nabla x_h)$ and $\hat{\sigma}_h$ just considered can yield an asymptotically exact error estimator, since the residual expression $\|\nabla \cdot \sigma_h + f\|_\Omega$ cannot be controlled properly. Moreover, the efficiency index of the estimator M_0 is most probably not even uniformly bounded. We may surmise, that the lacking efficiency of the estimator M_0 is caused by limitations of the ansatz, with the help of which we aim to approximate the optimal dual parameter. To show, that this is not the case, let us introduce a new vector field $\tilde{\sigma}_h \in (P_c^k(\mathcal{M}_h))^n$ by applying the canonical finite element interpolation $I_k : C(\Omega) \longrightarrow P_c^k(\Omega)$ to each coordinate of the field ∇x_0 in turn:

$$\tilde{\sigma}_h \quad := \quad \sum_{i=1}^{n} \; I_k \Big(\frac{\partial x_0}{\partial \xi_i} \Big) \; e_i \quad ,$$

assuming as before, that the analytical solution x_0 is contained at least in $H^{k+2}(\Omega)$. We conclude:

$$M_0(\tilde{\sigma}_h) \quad \leq \quad | x_h - x_0 |_{\Omega,1} \; + \; C \, \| \tilde{\sigma}_h - \nabla x_0 \|_{\Omega,1} \quad \leq \quad | x_h - x_0 |_{\Omega,1} \; + \; \mathcal{O}(h^k) \quad .$$

Solving the dual formulation

Clearly, any post-processing procedure which takes the gradient of the numerical approximation and generates a sufficiently smooth vector field $\sigma_h \in H_{\rm div}(\Omega)$ will suffer from the same drawback: Since the numerical approximation x_h need not be particularly accurate, no useful bound for the residual in the second duality relation can be guaranteed. To overcome this problem, an approximation to the optimal parameter σ_0, we have discussed in section 2.1.4, must be found directly. The simplest approach relies on a discretisation of the dual mixed formulation based on elements of Raviart-Thomas type. Let us seek a pair of functions $\{\sigma_h, u_h\} \in R_c^k(\mathcal{M}_h) \times P^k(\mathcal{M}_h)$, such that any set of test functions $\{\tau_h, y_h\} \in R_c^k(\mathcal{M}_h) \times P^k(\mathcal{M}_h)$ satisfies:

$$\left| \begin{array}{rcl} \langle \, \tau_h, \sigma_h \, \rangle_\Omega \; + \; (\, u_h, \nabla \cdot \tau_h \,)_\Omega & = & 0 \\ (\, y_h, \nabla \cdot \sigma_h \,)_\Omega & = & (-f, y_h \,)_\Omega \end{array} \right| \quad . \tag{3.40}$$

By exploiting the approximation properties (3.22) of the canonical finite element interpolation operator $\Pi : H_{\rm div}(\Omega) \longrightarrow R_c^k(\mathcal{M}_h)$ and by using basically the same analytical methods as those described in section 3.2.3 we can obtain the following a priori bounds:

$$\begin{array}{rcl} \| \nabla x_0 \, - \, \sigma_h \, \|_\Omega & \leq & C \;\; h^{k+1} \;\; | \, x_0 \, |_{\Omega,k+2} \\ \| \, \Delta \, x_0 \, - \, \nabla \cdot \sigma_h \, \|_\Omega & \leq & C' \;\; h^{k+1} \;\; | \, \Delta \, x_0 \, |_{\Omega,k+1} \end{array} \tag{3.41}$$

provided the data of the Dirichlet problem is sufficiently smooth: $f \in H^{k+1}(\Omega)$ for some $k \geq 0$. We note, that the residual expression $\nabla \cdot \sigma_h + f$ is orthogonal to all piecewise constant functions. Therefore, we can easily obtain a bound on this quantity in the dual norm:

$$| \, \nabla \cdot \sigma_h + f \, |_{\Omega,-1} \quad = \quad \sup_{v \in H_0^1(\Omega)} \frac{(\, \nabla \cdot \sigma_h + f, v - P_0 \, v \,)_\Omega}{| \, v \, |_{\Omega,1}} \quad \leq \quad C \; h \;\; \| \, \nabla \cdot \sigma_h + f \, \|_\Omega \quad .$$

Consequently, we may not only evaluate the hypercycle estimate (2.8) but also the sharper error bound (2.7), if we presume the numerical solution $\{\sigma_h, u_h\}$ of the dual formulation to be exact. We infer from (2.7) and (3.41), that we can consider the residual in the second duality relation as a higher order perturbation to the residual in the first duality relation. For by minimising the hypercycle estimate (2.7) with respect to the equilibration parameter κ we find:

$$| \, x_h - x_0 \, |_{\Omega,1} \quad \leq \quad M_{-1}(\sigma_h) \quad := \quad \| \, \nabla x_h \, - \, \sigma_h \, \|_\Omega \; + \; | \, \nabla \cdot \sigma_h \, + \, f \, |_{\Omega,-1} \quad .$$

The efficiency index of the above estimate is uniformly bounded from above. Still the estimator $M_{-1}(\sigma_h)$ is not asymptotically exact, unless we employ a Raviart-Thomas ansatz of higher order

$k > 0$ to approximate the solution of the dual formulation. The simplest ansatz for the dual variable σ_h which warrants the efficiency index to be asymptotically optimal, that means

$$\lim_{h\to 0} \frac{M_{-1}(\sigma_h)}{|\, x_0 - x_h \,|_{\Omega,1}} \quad = \quad 1 \quad ,$$

is based on the so called BDM element of lowest order $k = 1$ (for its properties and the etymology of its name we refer to [34]). The solution $\{\hat\sigma_h, \hat u_h\}$ of the dual mixed formulation (3.40) and the test functions are both contained in the space $B_c^k(\mathcal{M}_h) \times P^{k-1}(\mathcal{M}_h)$. The numerical solution satisfies the following a priori bounds:

$$\begin{aligned} \| \nabla x_0 \,-\, \hat\sigma_h \|_\Omega \quad &\le \quad C \;\; h^{k+1} \;\; |\, x_0 \,|_{\Omega,k+2} \\ \| \Delta\, x_0 \,-\, \nabla\cdot \hat\sigma_h \|_\Omega \quad &\le \quad C' \;\; h^{k} \;\; |\, \Delta\, x_0 \,|_{\Omega,k+1} \quad . \end{aligned}$$

Remark 3.11 The shape functions of Brezzi-Douglas-Marini elements contain full polynomial spaces in each component of the vector field. Still, a hypercycle estimate, which is based on some smoothed approximation $\sigma_h \in (P^k(\mathcal{M}_h))^n$ to the gradient ∇x_h, differs significantly from $M_0(\hat\sigma_h)$. Clearly, the dual solution is geared towards meeting the static admissibility constraint $\nabla\cdot\hat\sigma_h + f = 0$, while the averaged gradient aims at minimising the the residual in the first duality relation $\sigma = \nabla x$. Apart from this obvious dissimilarity, there is an important difference on the technical level: while the recovered gradient is continuous and therefore conforming, the requirement $\hat\sigma_h \in H_{\rm div}(\Omega)$ merely implies, that the normal components of the fluxes must be continuous across element interfaces. Since the tangential components may differ, the trial spaces $B^k(\mathcal{M}_h)$ have many more degrees of freedom than their counterparts $(P^k(\mathcal{M}_h))^n$.

Remark 3.12 To obtain a possibly small error bound (2.8), we should equilibrate the residual in the first and the residual in the second duality relation. Accordingly, it seems but reasonable to compute a conforming vector field σ_h, which is close to ∇x_h, and meets the requirement $\nabla\cdot\sigma_h + P_k f = 0$ for some index $k \ge 0$ at the same time. To determine such a field let us introduce a Lagrange multiplier $\mu_h \in P^k(\mathcal{M}_h)$ and define:

$$L(\sigma, \mu_h) \quad := \quad \frac{1}{2} \left\| \sigma_h - \nabla x_h \right\|_\Omega^2 \;\; - \;\; \left(\mu_h, \nabla\cdot\sigma_h + f \right) \quad ; \quad \sigma \in H_{\rm div}(\Omega) \;\; .$$

Whether an unique solution exists for the associated saddle point problem, depends on our choice of the ansatz for the dual variable. If we assume for instance $\hat\sigma_h \in B_c^{k+1}(\Omega)$, the saddle point will be described by the following linear system, valid for any $\{\tau_h, y_h\} \in B_c^{k+1}(\Omega) \times P^k(\mathcal{M}_h)$:

$$\left| \begin{array}{rcrcl} \langle\, \tau_h, \hat\sigma_h - \nabla x_h \,\rangle_\Omega & + & (\, \hat\mu_h, \nabla\cdot\tau_h \,)_\Omega & = & 0 \\ (\, y_h, \nabla\cdot\sigma_h \,)_\Omega & & & = & (\, -f, y_h \,)_\Omega \end{array} \right| \quad .$$

We note, that the above system takes the very form of (3.40), if we perform a partial integration to remove the gradient from the numerical solution $x_h \in H_0^1(\Omega)$ and define $u_h := \hat\mu_h + P_k\, x_h$. Hence, our idea of finding the dual parameter σ_h by a least squares approach has simply led us back to solving the dual formulation of the Laplace problem.

Remark 3.13 Though trial spaces of Raviart-Thomas type are well established as a means of obtaining a discrete formulation of the Laplace problem, we are not bound to use them. The ideas put forward in the previous remark may prompt us to employ Lagrange elements for the dual parameter σ_h instead of discontinuous shape functions. However, unless we dismiss the static admissibility constraint $\nabla\cdot\sigma_h + P_k f = 0$ completely, we must ensure, that the compatibility condition for our trial spaces is met. While we can ignore the Lagrange multiplier μ_h for our purposes, we must warrant the consistency of the saddle point problem. The setting is quite similar to the statement of the Stokes problem except for the fact, that the incompressibility constraint is per se consistent. A very general theory for the numerical treatment of saddle point formulations is presented in [112]. A thorough exposition of the topic is given in §2 of [35] along with a number of examples in §6, how to construct pairs of compatible trial spaces for the vector field σ_h and the Lagrange multiplier μ_h. Unfortunately, nonconforming elements of Crouzeix-Raviart type [54] cannot be used to approximate the components of the dual parameter $\sigma_h \in H_{\rm div}(\Omega)$, since the divergence of the resulting fields is not square integrable.

3.3.2 A posteriori estimates for the Obstacle Problem

In the following we will use the notation we have introduced in section 2.2. To simplify our exposition, let us assume, that the tensor $A \in W^{1,\infty}(\Omega, \mathbb{R}^{n\times n})$ is the identity matrix $I \in \mathbb{R}^{n\times n}$ and that the set E, which describes the compass of the obstacle is identical to the computational domain Ω, a convex set with a polygonal boundary. The shape of the obstacle shall be determined by some function $\psi \in W^{2,p}(\Omega)$, which has the property $\Delta\psi \in L^p(\Omega)$ for some exponent $p > n$ and is negative along $\partial\Omega$. If the load function satisfies $f \in L^p(\Omega)$, the solution of the obstacle problem stated in section 2.2.2 is known (see e. g. [33, 101]) to be continuously differentiable with the Hölder exponent $\alpha = 1 - n/p$. To be more specific, we find: $x_0 \in W^{2,p}(\Omega)$.

Even if we suppose the function $f \in W^{1,1}(\Omega) \cap L^\infty(\Omega)$ to be slightly more regular and the obstacle $\psi \in C^3(\Omega)$ to be smooth, the regularity of the solution x_0 will in general not exceed $x_0 \in W^{s,p}(\Omega)$ with $s < 2 + 1/p$ being an arbitrary parameter [32]. Since this lack of smooth solutions is an intrinsic feature of the obstacle problem, we can not hope to evade its impact by transforming the variational formulation. In consequence, we are unable to turn the computable hypercycle estimates (2.16) and (2.19) into asymptotically exact error bounds simply by raising the polynomial degree of the shape functions with which we construct the parameter $\sigma \in H_{\text{div}}(\Omega)$. If we choose to ignore this difficulty, our options of constructing suitable dual parameters are the same as in the case of the Laplace problem: We may smooth the gradient ∇x_h of the primal approximation or solve the dual formulation of the obstacle problem numerically. Furthermore, we may attempt to minimise the hypercycle estimate with respect to the dual parameter. This latter approach we can of course also combine with one of the other methods to provide a good initial guess for the optimisation algorithm.

Solving the dual formulation

In section 4.3 we will examine those multilevel methods we employ to solve the algebraic problems which arise out of a finite element discretisation of the obstacle problem in its dual formulation. We are going to demonstrate in section 4.3.2 how to derive that formulation. Hence, let us merely state in the present context, that we must seek a vector field $\sigma_h \in R_c^k(\mathcal{M}_h)$, which is admissible

$$\sigma_h \;\in\; S_h(f) \;:=\; \left\{\, \tau_h \in R_c^k(\mathcal{M}_h) \;\middle|\; \nabla\cdot \tau_h \,+\, f \,\leq\, 0 \text{ a. e. on } \Omega \,\right\}$$

and which satisfies the following variational inequality:

$$\langle\, \sigma_h, \tau_h - \sigma_h \,\rangle_\Omega \;+\; \langle\, \psi, \nabla\cdot \tau_h - \nabla\cdot \sigma_h \,\rangle_\Omega \;\geq\; 0 \quad ; \quad \tau_h \in S_h(f) \;.$$

If we use an Raviart-Thomas ansatz of low order $k \in \{0, 1\}$ or alternatively base our discretisation on the BDM element of low order (shifting the index k down by 1 for that purpose) we may enforce the admissibility constraint $\sigma_h \in S_h(f)$ by testing with the nonnegative shape functions ψ_i of the trial space $P^k(\mathcal{M}_h)$. To facilitate such a device we must replace the forcing function $f \in L^2(\Omega)$ by its image under the projection operator $P_k : L^2(\Omega) \longrightarrow P^k(\mathcal{M}_h)$ defined by (3.23). If $\mathfrak{I}_k$ denotes the set of all degrees of freedom featured by the space $P^k(\mathcal{M}_h)$, we may state:

$$(\, \psi_i, \nabla\cdot \sigma_h + f \,)_\Omega \;\leq\; 0 \quad ; \quad i \in \mathfrak{I}_k \qquad \Longleftrightarrow \qquad \sigma_h \,\in\, S_h(P_k f) \;.$$

We remark, that the above would no longer hold, if we employed trial spaces of higher order. However, due to the limited regularity of the analytic solution $\sigma_0 \in W^{1+r,2}(\Omega, \mathbb{R}^n)$ with $r < 0.5$ such a choice would be questionable in any case. The resulting algebraic formulation exhibits the structure of a saddle point problem and features in addition a linear complementary condition for the Lagrange multiplier $u_h \in P^k(\mathcal{M}_h)$ associated with the admissibility constraint:

$$u_h \,\geq\, 0 \qquad \wedge \qquad \nabla\cdot \sigma_h \,+\, P_k f \;\leq\; 0 \qquad \wedge \qquad (\, u_h, \nabla\cdot \sigma_h + f \,)_\Omega \;=\; 0 \;. \tag{3.42}$$

We are not aware of any published theoretical results, which assert, that the Lagrange multiplier u_h approximates the distance $x_0 - P_k\psi$ between the analytical solution of the primal formulation and the obstacle, or rather its projection into the trial space $P^k(\mathcal{M}_h)$ in a quasi optimal sense:

$$\|\, u_h + \psi - x_0 \,\|_{\infty,\Omega} \;\leq\; C\, h^{k+1}\, (\log h)^2 \;\, |\, x_0 \,|_{\Omega,2} \;\;,$$

though similar approximation results [114] have been obtained for the primal formulation of the penalised obstacle problem. Under certain assumptions on the properties of the solution x_0 near the boundary of the coincidence set Ω_0 (see again [114] and the references cited therein) we can derive from such estimates bounds on the measure of the symmetric difference

$$\Omega_0 \div \Omega_h^x \quad := \quad \Omega_0 \setminus \Omega_h^x \quad \cup \quad \Omega_h^x \setminus \Omega_0$$

between the coincidence set of the analytical solution x_0 and the coincidence set Ω_h^x of its primal approximation. If we denote by Ω_h^u the coincidence set associated with the Lagrange multiplier u_h, we may therefore surmise for $k \in \{0, 1\}$ an estimate of the form:

$$| \Omega_h^x \div \Omega_h^u | \quad \leq \quad C \sqrt{h^{k+1}} \; | \log h | \quad . \tag{3.43}$$

In the following let us suppose, that (3.43) holds. The computable hypercycle estimate (2.15) may now be bounded by exploiting the linear complementary condition (3.42). Let us assume, the forcing function is sufficiently smooth to warrant an analytical solution $x_0 \in W^{2,\infty}(\Omega)$. We select the parameter $\kappa^* > 0$ in the very same way, we have done in the previous section for the unconstrained problem, take the square root and thus arrive at the estimate:

$$| x_h - x_0 |_{\Omega,1} \quad \leq \quad \| \nabla x_h - \sigma_h \|_\Omega \; + \; | f - P_k f |_{\Omega,-1} \; + \; \lambda_0^{-1} \inf_{\mu \in L^2(\Omega) \cap \mathfrak{M}_x} \| \mu + P_k f + \nabla \cdot \sigma_h \|_\Omega \quad .$$

On the complement $\Omega \setminus \Omega_0^u$ of the "dual" coincidence set the residual expression $P_k f + \nabla \cdot \sigma_h$ vanishes, whereas any negative contribution by said function can be compensated on the "primal" coincidence set Ω_0^x by the positive measure μ. In consequence, the integrand can be negative only on the set $\Omega^- := (\Omega \setminus \Omega_0^x) \cap \Omega_0^u$. Since the residual is uniformly bounded, we conclude:

$$\inf_{\mu \in L^2(\Omega) \cap \mathfrak{M}_x} \| \mu + P_k f + \nabla \cdot \sigma_h \|_\Omega \quad \leq \quad \| P_k f \; + \; \nabla \cdot \sigma_h \|_{\infty,\Omega^-} \; | \Omega_h^x \div \Omega_h^u | \quad .$$

Unfortunately, the residual expression $P_k f + \nabla \cdot \sigma_h$ is not continuous across element boundaries. Hence, we are unable to control the supremum norm over the set Ω^- by some power of the mesh parameter h, even though the residual vanishes on certain of the adjoining elements.

Remark 3.14 Previously, we have considered an hypercycle estimate, which has involved the evaluation of some dual norm. Strictly speaking, such an estimate is not computable. However, we have found the contribution of the residual in the second duality relation to be a higher order perturbation, if that quantity is but measured in the dual norm. Consequently, we may justify our deliberations by the realisation, that we can simply ignore the second part of the hypercycle estimate and still get a reasonable if not thoroughly reliable error bound. The same observation holds true, if we compute the error estimate (2.8) using a trial space of higher accuracy for the dual parameter than for the primal approximation. For example, if we have obtained a numerical solution $x_h \in P_c^1(\mathcal{M}_h)$, we may solve the dual formulation with the ansatz $R_c^1(\mathcal{M}_h)$. As far as the obstacle problem is concerned, the situation is different: The residual contribution in the second duality relation forms a substantial part of the error majorant, which we must not neglect, even if we have employed higher order trial spaces to represent the dual parameter σ_h. Moreover, with a view to (3.43) we cannot warrant a mesh independent bound on the efficiency index of our hypercycle estimate. Hence, it seems paramount to match the boundary of the coincidence set Ω_0 as closely as possible. Since the primal approximation $x_h \in P_c^1(\mathcal{M}_h)$ satisfies an estimate of the form (3.43) with $k = 1$ and Ω_h^u replaced by Ω_0, the efficiency index will behave under uniform mesh refinement like $C\,|\log h|$ provided our assumptions on the approximation properties of the dual approximation $u_h \in P^k(\mathcal{M}_h)$ are correct and either $k = 1$ holds true or the dual parameter is computed on a different mesh $\mathcal{M}_h'$, which we have locally refined in some neighbourhood around the boundary of the coincidence set. If the local mesh width h' satisfies the requirement $h' < h^2$, we may hope to recover an almost efficient error bound.

Remark 3.15 The above remark suggests a procedure how to locally adapt meshes in order to improve both the accuracy of the primal approximation and the efficiency of the a posteriori error estimate. As long as the residual in the second duality relation can be considered a higher

order perturbation, we may obviously employ the residual in the first duality relation as a local error indicator. Since that latter residual is a simple integral, we can decompose it directly into element contributions and thus localise the a posteriori error estimate:

$$2\, M_F(\nabla x_h, \sigma_h) \;=\; \int\limits_\Omega (\,\nabla x_h - \sigma_h\,)^2\, d\,\xi \;=\; \sum_{M\in\mathcal{M}_h} \|\,\nabla x_h - \sigma_h\,\|^2_{M} \;=:\; \sum_{M\in\mathcal{M}_h} \eta^2_{M}\;.$$

The mesh $\mathcal{M}_h$ on which we calculate the primal approximation may now be refined in such a way, that the local error indicators η_{M} become equilibrated. The second mesh $\mathcal{M}'_h$ which supplies the dual parameter σ_h, must take into account possible breakdowns in the regularity of the analytical solution, which cause the second residual contribution $\|\nabla\cdot\,\sigma_h + f\|_{\Omega}$ to deteriorate. Hence, it seems but reasonable to refine the mesh $\mathcal{M}'_h$, wherever the local indicators $\zeta_{M} > 0$, defined by

$$\big\|\,\nabla\cdot\,\sigma_h \,+\, f\,\big\|^2_{\Omega} \;-\; \big\|\,0\,\wedge\,\big(\,\nabla\cdot\,\sigma_h \,+\, f\,\big)\,\big\|^2_{\Omega_x} \qquad =: \qquad \sum_{M\in\mathcal{M}'_h} \zeta^2_{M}\;,$$

are large. In actual computations, the mesh $\mathcal{M}'_h$ for the dual parameter can be adapted along with the mesh, on which the approximation of the primal formulation is obtained. The elements to be refined are selected based on the information provided by the local indicators η_{M} and ζ_{M}. After the mesh refinement has been carried out and a new primal approximation has been computed, a third mesh $\mathcal{M}''_h$ may be constructed, which contains all the edges and all the vertices, which are present either in $\mathcal{M}_h$ or in $\mathcal{M}'_h$. On this new mesh the hypercycle estimate can be evaluated without incurring any interpolation errors, that may bias the refinement procedure.

On the properties of the alternative error estimate

Analysing the alternative hypercycle estimate (2.19) which we have developed in section 2.2.5 is rendered even more difficult by the fact, that we can not eliminate the dependence of the error bound on the equilibration parameter κ as we have been able to do in the previous paragraphs. In the following we will therefore merely collect some remarks we have already presented in a similar form in [41]. Let us begin our exposition with a technical lemma:

Lemma 3.1 *The function* $\Theta : \mathbb{R}\times\mathbb{R} \longrightarrow \mathbb{R}$, *defined by*

$$\Theta(r,s) \;=\; \begin{cases} s^2 & ; \quad r\,+\,s\,\geq\,0 \\ s^2\,-\,(\,r\,+\,s\,)^2 & ; \quad \textit{else} \end{cases}\;,$$

is continuously differentiable, concave with respect to its first and convex with respect to its second argument. Furthermore, the following monotonicity property holds:

$$0 \;\leq\; r \quad\Rightarrow\quad 0 \;\leq\; r\,\frac{\partial\Theta}{\partial r}(r,s) \;\leq\; \Theta(r,s) \quad ; \quad s\,\in\,\mathbb{R}\;. \tag{3.44}$$

Proof The above results can be obtained by elementary calculus. □

We note, that we can express the hypercycle estimate (2.19) as an integral involving the very function Θ, we have introduced in the above lemma, as a kernel. Let us define:

$$M^{(\kappa)}_{\Theta}(x,\sigma) \quad := \quad \frac{1+\kappa}{\lambda_0^2}\,\int\limits_\Omega \Theta\Big(\,\lambda_0^2\,\frac{x-\psi}{1+\kappa}\,,\; f\,+\,\nabla\cdot\,\sigma\,\Big)\; d\,\xi \qquad ; \qquad x\,\in\,V_\psi\;.$$

Using the same notation as in section 1.2.7 we may now state for any vector field $\sigma \in H_{\mathrm{div}}(\Omega)$:

$$\tilde{M}_K(x,\sigma) \;=\; \frac{\kappa+1}{\kappa}\; M_F(\nabla x,\sigma) \;+\; \frac{1}{2}\; M^{(\kappa)}_{\Theta}(x,\sigma) \quad . \tag{3.45}$$

From the above lemma and from (3.45) we infer, the generalised hypercycle estimate (2.19) can never become negative, as long as $x \geq \psi$ holds. In addition we find the convexity of $M^{(\kappa)}_{\Theta}(\,\cdot\,,x)$ and the concavity of $M^{(\kappa)}_{\Theta}(\sigma,\cdot\,)$ resulting from the properties of the kernel Θ. We claim:

Proposition 3.11 *The residual part $M_\Theta^{(\kappa)} : V_\psi \times H_{\mathrm{div}}(\Omega) \longrightarrow \mathbb{R}_0^+$ of the generalised hypercycle estimate (2.19) is a continuous functional, that is convex with respect to its first and concave with respect to its second argument. It is differentiable and monotonously increasing in $\kappa \geq 0$.*

Proof For simplicity, let us identify the expression $\lambda_0^{-2}(1+\kappa)$ with the parameter κ. When we combine the lemma 3.1 with the our integral representation of $M_\Theta^{(\kappa)}$ and apply first Hölder's and then Minkowsky's inequality, we obtain for arbitrary functions $x, \hat{x} \in V_\psi$ and arbitrary fields $\sigma, \hat{\sigma} \in H_{\mathrm{div}}(\Omega)$ the following estimate:

$$\begin{aligned} \left| M_\Theta^{(\kappa)}(x,\sigma) - M_\Theta^{(\kappa)}(\hat{x},\hat{\sigma}) \right| &\leq 2\left(\kappa \,\|\, \nabla\cdot\sigma + f \,\|_\Omega + \kappa \,\|\, \nabla\cdot\hat{\sigma} + f \,\|_\Omega + \|\, x - \psi \,\|_\Omega + \right. \\ &+ \left. \|\, \hat{x} - \psi \,\|_\Omega \right) \left(\|\, \nabla\cdot\sigma - \nabla\cdot\hat{\sigma} \,\|_\Omega + \|\, \kappa(x-\hat{x}) \,\|_\Omega \right) . \end{aligned}$$

Hence, $M_\Theta^{(\kappa)}$ is continuous in the norm $\|\!|\cdot|\!\|_\Omega$. Let us fix $b \in L^2(\Omega)$ and $g \in L^2(\Omega, \mathbb{R}_0^+)$. Since the kernel Θ is continuously differentiable, we can compute the partial derivative of $\Theta(\kappa^{-1}g, b)$ with respect to $\kappa \geq \lambda_0^{-2} > 0$ almost everywhere on Ω. From the inequality (3.44) we infer:

$$\left| \frac{\partial\Theta}{\partial\kappa}\big(\kappa^{-1}g, b\big) \right| = \frac{\partial\Theta}{\partial r}\big(\kappa^{-1}g, b\big)\,\frac{g}{\kappa^2} \leq \frac{1}{\kappa}\,\Theta\big(\kappa^{-1}g, b\big) \in L^1(\Omega) .$$

Lebesgue's theorem of dominated convergence implies, that the partial derivative of the integral in the definition of $M_\Theta^{(\kappa)}$ with respect to κ exists. Therefore, we can employ the inequality (3.44) again to find a lower bound on this quantity for any $x \in V_\psi$ and any $\sigma \in H_{\mathrm{div}}(\Omega)$:

$$\begin{aligned} \lambda_0^2 \frac{\partial}{\partial\kappa} M_\Theta^{(\kappa)}(x,\sigma) &= -\kappa \int_\Omega \frac{\partial\Theta}{\partial r}\big(\,\kappa^{-1}(x-\psi)\,,\, f + \nabla\cdot\sigma\,\big)\,\frac{x-\psi}{\kappa^2}\; d\xi \\ &+ \int_\Omega \Theta\big(\,\kappa^{-1}(x-\psi)\,,\, f + \nabla\cdot\sigma\,\big)\; d\xi \quad \geq \quad 0 . \end{aligned}$$

The algebraic properties of $M_\Theta^{(\kappa)}$ stem from those of the kernel Θ, we have defined in lemma 3.1, and can be verified by elementary algebraic manipulations. □

The above results can be used to provide an upper bound for the hypercycle estimate $\tilde{M}_K$, when the dual parameter σ is but contained in a finite dimensional subspace of $H_{\mathrm{div}}(\Omega)$. Before we proceed, let us introduce for any subset $Q \subset H_{\mathrm{div}}(\Omega)$ the sharpest possible estimate, we can obtain by minimising the a posteriori error bound (2.19) with respect to $\sigma \in Q$. We define:

$$\mathcal{H}^{(\kappa)}(x, Q) := \inf_{\sigma\in Q} \left\{ \frac{\kappa+1}{\kappa} \,\|\, \nabla x - \sigma \,\|_\Omega^2 + M_\Theta^{(\kappa)}(x,\sigma) \right\} .$$

Considering the case $Q = H_{\mathrm{div}}(\Omega)$, we find by proposition 2.1, that the minimiser $\sigma_\kappa \in Q$ of the right hand side is related to the gradient of an arbitrary function $x \in H_0^1(\Omega)$ by the estimate:

$$\|\, \nabla x - \sigma_\kappa \,\|_\Omega \leq \frac{4+\kappa}{1+\kappa} \,|\, x - x_0 \,|_{\Omega,1} .$$

For reasons, that will become apparent below, let us fix a constant $\alpha \in (0,1)$ and introduce a new equilibration parameter $\hat{\kappa}$, larger than κ, which shall be defined by the formula:

$$\hat{\kappa} := \frac{\kappa + \alpha}{1 - \alpha} .$$

Let $\sigma \in Q$ denote an arbitrary field. We may now control the error bound $\mathcal{H}^{(\kappa)}(x,Q)$ by:

$$\begin{aligned} \mathcal{H}^{(\kappa)}(x,Q) &\leq \frac{1+\kappa}{\kappa}\left\{ (1+\hat{\kappa}) \,\|\, \sigma - \sigma_{\hat{\kappa}} \,\|_\Omega^2 + \frac{1+\hat{\kappa}}{\hat{\kappa}} \,\|\, \nabla x - \sigma_{\hat{\kappa}} \,\|_\Omega^2 \right\} + M_\Theta^{(\kappa)}(x,\sigma) \\ &\leq \frac{(1+\kappa)^2}{\kappa\,(1-\alpha)} \,\|\, \sigma - \sigma_{\hat{\kappa}} \,\|_\Omega^2 + \frac{(4-3\,\alpha+\kappa)^2}{\kappa\,(1+\kappa)(\alpha+\kappa)} \,|\, x - x_0 \,|_{\Omega,1}^2 + M_\Theta^{(\kappa)}(x,\sigma) \\ &+ \left\{ \frac{1+\hat{\kappa}}{\hat{\kappa}} \,\|\, \nabla x - \sigma_{\hat{\kappa}} \,\|_\Omega^2 + M_\Theta^{(\hat{\kappa})}(x,\sigma_{\hat{\kappa}}) \right\} - M_\Theta^{(\hat{\kappa})}(x,\sigma_{\hat{\kappa}}) . \end{aligned} \tag{3.46}$$

Thanks to the proposition 2.2 we can bound the expression in brackets in terms of the sharpest estimate (2.27) for the energy error which is available to us. The remaining expressions either depend on the approximation error or can be controlled in terms of the distance $\| \sigma - \sigma_{\hat{\kappa}} \|_\Omega$ as we are now going to show. For any two fields $\sigma, \hat{\sigma} \in H_{\rm div}(\Omega)$ and any $x \in V_\psi$ we find:

$$
\begin{aligned}
M_\Theta^{(\kappa)}(\sigma, x) &= \frac{1+\kappa}{\lambda_0^2} \int_\Omega \Theta\Big(\lambda_0^2 \, \frac{x-\psi}{1+\kappa} \,, \;\; f + \nabla\cdot\sigma \Big) \; dx \\
&= \frac{1+\kappa}{\lambda_0^2} \int_\Omega \Theta\Big(\lambda_0^2 \, \frac{x-\psi}{1+\kappa} \,, (1-\alpha) \, \frac{f+\nabla\cdot\sigma}{1-\alpha} + \alpha \, \frac{\nabla\cdot\sigma - \nabla\cdot\hat{\sigma}}{\alpha} \Big) \, d\xi \\
&\leq \frac{(1-\alpha)\,(1+\kappa)}{\lambda_0^2} \int_\Omega \Theta\Big(\lambda_0^2 \, \frac{(1-\alpha)\,(x-\psi)}{(1+\kappa)\,(1-\alpha)} \,, \; \frac{f+\nabla\cdot\hat{\sigma}}{1-\alpha} \Big) \; d\xi \quad + \\
&+ \frac{\alpha\,(1+\kappa)}{\lambda_0^2} \int_\Omega \Theta\Big(\lambda_0^2 \, \frac{\alpha\,(x-\psi)}{(1+\kappa)\,\alpha} \,, \; \frac{\nabla\cdot\sigma - \nabla\cdot\hat{\sigma}}{\alpha} \Big) \; d\xi \\
&= \frac{1}{\lambda_0^2} \, \frac{1+\kappa}{1-\alpha} \int_\Omega \Theta\Big(\lambda_0^2 \, \frac{(1-\alpha)\,(x-\psi)}{1+\kappa} \,, \; f + \nabla\cdot\hat{\sigma} \Big) \; d\xi \quad + \\
&+ \frac{1+\kappa}{\alpha\,\lambda_0^2} \int_\Omega \Theta\Big(\lambda_0^2 \, \frac{\alpha\,(x-\psi)}{1+\kappa} \,, \; \nabla\cdot\sigma - \nabla\cdot\hat{\sigma} \Big) \; d\xi \\
&\leq M_\Theta^{(\hat{\kappa})}(x, \hat{\sigma}) + \frac{1+\kappa}{\alpha\,\lambda_0^2} \, \| \nabla\cdot\hat{\sigma} - \nabla\cdot\sigma \|_\Omega^2 \quad .
\end{aligned}
$$

To simplify the notation let us introduce the following abbreviations:

$$
\begin{aligned}
C_0(\alpha, \kappa, \sigma, \hat{\sigma}) &:= \frac{1+\kappa}{\kappa\,(1-\alpha)} \, \| \sigma - \hat{\sigma} \|_\Omega^2 + \frac{1}{\alpha\,\lambda_0^2} \, \| \nabla\cdot\sigma - \nabla\cdot\hat{\sigma} \|_\Omega^2 \;, \\
C_1(\alpha, \kappa) &:= \frac{(\,4 - 3\,\alpha + \kappa\,)^2}{(\,1+\kappa\,)\,(\,\kappa+\alpha\,)} \quad .
\end{aligned}
$$

Both of these expressions stay bounded in the limit $\kappa \to \infty$, whereas at least C_0 is divergent in the case $\kappa \to 0$. The behaviour of $C_1(\alpha, \kappa)$ depends on our choice of the constant $\alpha \in (0, 1)$. In the limit $\alpha = 1$ we find $C_1(1, \kappa) \equiv 1$, which would result in the tightest bound for the efficiency index of the error estimate $\mathcal{H}^{(\kappa)}(x, Q)$, that is available to us after our treatment of that quantity in (3.46). Unfortunately, C_h is not defined for $\alpha = 1$, whence it will be necessary to balance the size of C_0 against that of C_1. Combining the above results we infer:

$$
\mathcal{H}^{(\kappa)}(x, Q) \;\leq\; 2 \, \frac{1+\kappa}{\alpha+\kappa} \, \big\{ J(x) - \inf_{v \in V_\psi} J(v) \big\} + \frac{C_1(\alpha, \kappa)}{\kappa} \, | \, x - x_0 \, |_{\Omega,1}^2 + (1+\kappa) \inf_{\tau \in Q} C_0(\alpha, \kappa, \sigma_{\hat{\kappa}}, \tau) \; .
$$

The solution of the perturbed obstacle problem, we have discussed in section 2.2.8, is but of reduced regularity just as the solution of the original problem (2.12). In consequence, we find $\sigma_{\hat{\kappa}} \in H^s(\Omega, \mathbb{R}^n)$ with $s < 1.5$ and therefore our options in minimising the quantity C_0 are limited. The best approximation results possible are already obtained, if we employ a product ansatz of the form $Q = (P_c^1(\mathcal{M}_h))^n$ or alternative use BDM-elements of lowest order: $Q = B_c^1(\mathcal{M}_h)$. In both cases there will be constants $C, C' > 0$ which will only depend on the data of the obstacle problem and not on $\hat{\kappa}$, such that the following estimate holds:

$$
\inf_{\tau \in Q} C_0(\alpha, \kappa, \sigma_{\hat{\kappa}}, \tau) \quad \leq \quad h^{2s-2} \; \Big\{ \; \frac{1+\kappa}{\kappa\,(1-\alpha)} \, C \, h^2 \; + \; \frac{1}{\alpha} \, C' \; \Big\} \quad . \tag{3.47}
$$

More accurate results can only be obtained with the help of local mesh refinements concentrated in a neighbourhood around the boundary of the coincidence set. We may infer from (3.47), that

we can choose the parameter α as large as $1 - a\,h^2$ for some fixed scaling factor $a > 0$ without our impairing the asymptotic order of the resulting upper bound. We conclude:

Proposition 3.12 *If $\Omega \subset \mathbb{R}^n$ is a domain with a sufficiently smooth boundary or alternatively convex, $f \in W^{1,1}(\Omega) \cap L^\infty(\Omega)$ and $\psi \in H^3(\Omega)$, such that the solution x_0 of the obstacle problem is contained in $W^{2,\infty}(\Omega)$, a finite element approximation scheme based on BDM-elements of lowest order or alternatively a product ansatz of the form $Q = (P_c^1(\mathcal{M}_h))^n$ yields for every admissible, bounded and sufficiently regular function $x \in V_\psi \cap W^{1,1}(\Omega) \cap L^\infty(\Omega)$ a generalised hypercycle estimate $\mathcal{H}^{(\kappa)}(x, Q)$, that can be controlled for any $\epsilon \in (0,1)$ and any $\kappa \geq \kappa_o > 0$ by:*

$$\mathcal{H}^{(\kappa)}(x,Q) \;\leq\; (2 + \mathcal{O}(h^2))\,(J(x) - J(x_0)) \;+\; \frac{1 + \mathcal{O}(h^2)}{\kappa}\,|\,x - x_0\,|^2_{\Omega,1} \;+\; (1+\kappa)\,C\,h^{1-\epsilon}\;.$$

The constant $C > 0$ depends on the data of the obstacle problem, on the approximate solution $x \in V_\psi$ and on the choice of ϵ and κ_o, but not on the parameter κ itself.

Proof The proof has been developed in the course of this subsection. □

We find the above approach slightly unsatisfactory in dealing with Raviart-Thomas elements of lowest order, since the approximation result (3.47) does not apply to them. When we construct a dual parameter σ contained in the trial space $Q := R_c^0(\mathcal{M}_h)$, we must replace it by:

$$\inf_{\tau \in Q} C_0(\alpha, \kappa, \sigma_{\hat\kappa}, \tau) \quad\leq\quad \Big\{\; \frac{1+\kappa}{\kappa\,(1-\alpha)}\,C\,h^2 \;+\; \frac{1}{\alpha}\,C'\,h^{2s-2} \;\Big\} \quad . \tag{3.48}$$

Consequently, we can merely choose the parameter α as large as $1 - a\,h$ without our affecting the asymptotic behaviour of the expression C_0. The resulting upper bound on the hypercycle estimate $\mathcal{H}^{(\kappa)}(x, Q)$ is of the same form as the one presented in proposition 3.12. However, the asymptotic order of the first two perturbation terms is dropped by one. As these terms can no longer be ignored when compared against the last contribution, which stems from the approximation of the divergence $\nabla\cdot\,\sigma_{\hat\kappa} \in H^{s-1}(\Omega)$, the proposition 3.12 would suggest a global refinement of the mesh, on which the parameter σ is computed, in order to raise the efficiency of the error bound. A slightly different result can be obtained, if we replace (3.46) by the following estimate:

$$\begin{aligned}
\mathcal{H}^{(\kappa)}(x,Q) \quad &\leq \quad \frac{1+\kappa}{\kappa}\,\Big\{\;(1+\kappa)\,\|\,\sigma - \sigma_{\hat\kappa}\,\|^2_\Omega \;+\; \frac{1+\kappa}{\kappa}\,\|\,\nabla x - \sigma_{\hat\kappa}\,\|^2_\Omega\;\Big\} \;+\; M^{(\kappa)}_\Theta(x,\sigma) \\
&\leq \quad \frac{(1+\kappa)^2}{\kappa}\,\|\,\sigma - \sigma_{\hat\kappa}\,\|^2_\Omega \;+\; \frac{(\,4 - 3\,\alpha + \kappa\,)^2}{(\,1+\kappa\,)^2}\,\Big(\,\frac{(1+\kappa)^2}{\kappa^2} \;-\; 1\,\Big)\,|\,x - x_0\,|^2_{\Omega,1} \\
&+ \quad 2\,\frac{1+\kappa}{\alpha+\kappa}\,\Big(\,J(x) - \inf_{w \in V_\psi} J(w)\,\Big) \;+\; \frac{1+\kappa}{\alpha\,\lambda_0^2}\,\|\,\nabla\cdot\,\sigma_{\hat\kappa} \;-\; \nabla\cdot\,\sigma\,\|^2_\Omega \quad .
\end{aligned}$$

In the limit $\alpha \to 1$ all the constants involved in the above estimate stay bounded. Moreover, we find $\sigma_{\hat\kappa} \to \nabla x_0$, while $\hat\kappa$ tends to ∞. Therefore, an approximation result similar to (3.48) holds except for the fact, that the scaling factors are now independent of the parameter α. We claim:

Proposition 3.13 *We assume, the data of obstacle problem satisfies the regularity assumptions formulated in proposition 3.12. If we choose $Q := R_c^0(\mathcal{M}_h)$, we can bound the hypercycle estimate $\mathcal{H}^{(\kappa)}(x, Q)$ for any $\epsilon \in (0,1)$ and any equilibration parameter $\kappa \geq \kappa_o > 0$ by:*

$$\mathcal{H}^{(\kappa)}(x,Q) \quad\leq\quad 2\,\frac{(\,1+\kappa\,)^2}{\kappa^2}\,\Big(\,J(x) \;-\; J(x_0)\,\Big) \;+\; (1+\kappa)\,C\,h^{1-\epsilon}\;.$$

The generic constant $C > 0$ depends on the data of obstacle problem, on the function $x \in V_\psi$ and on the choice of ϵ and κ_o, but not on the parameter κ itself.

Proof The above result is an easy consequence of the proposition 2.2 in the limit $\alpha \to 1$. □

Chapter 4

On the Evaluation of Duality Error Majorants and related Computational Issues

The previous chapters have served to establish a theory for a posteriori error computions which may be applied to a wide class of uniformly convex variational problems. The theory has been applied to some generic test problems and various issues which are connected to the discretisation both of the primal and of the dual formulation have been addressed. However, the success of a numerical technique is more often secured by slight technical advantages than by the most solid or profound theoretical foundation. Hence, we deem it necessary to supply some information on how we have implemented our numerical technology. Clearly, we cannot produce a manual for the finite element package we have developed as a prerequisite for our research: Due to the complexity of the software any ambition to furnish a technical reference must be futile. To effect a compromise we have written an appendix in which we describe the grammar of the finite element compiler. Anyone with a modicum of programming experience should be able to understand the library's interface and write finite element scripts once he has consulted the library's source code and said appendix. We may relate, however, how we have solved certain technical problems which may be seen pivotal to the succes of the numerical technology as a whole. As a matter of course, we do not represent that our solutions are the very best human ingenuity can possibly devise: We simply hope to demonstrate by our expositions that the numerical technology necessary to use a posteriori error estimators based on duality arguments is managable indeed.

4.1 General Remarks on Mesh Handling

The computation of a posteriori error estimates for the numerical solution of a partial differential equation is a sensible undertaking only if we can exploit our information about the approximation error either to increase the accuracy of our numerical results or to reduce the numerical complexity of the discrete problem. If we were limited to carrying out our calculations on regular grids, we could use a priori estimates to determine the optimal mesh width in advance. Hence, the empirical study of a posteriori estimates basically requires a finite element technology, that is capable of managing unstructured, locally refined meshes and of changing the mesh topology in accordance with the data of the a posteriori error estimator. The handling of locally refined meshes poses a number of difficulties, that are alien to finite element implementations that work only on structured or block structured meshes. In the following we will address some of the technical issues we have been forced to resolve in order to facilitate dynamic mesh adaption: We will review the data structures we have implemented and discuss algorithms for mesh refinement as well as coarsening. Since our approach to local mesh adaption relies on the concept of *hierarchical meshes* we must provide either special refinement or special discretisation schemes to interface such domains of the mesh which feature a higher level of refinement with those domains which do not. In the last paragraph of this section we will touch on possible solutions, including auxiliary refinement patterns. A detailed account of the algorithm we have employed in our finite element package to control auxiliary mesh refinements shall be deferred to section 4.2.4.

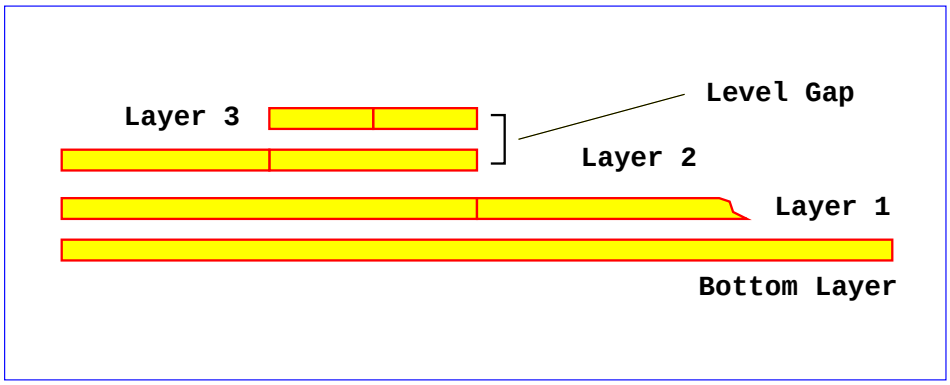
4.1.1 Technical Prerequisites for Adaptive Mesh Refinement

As we have seen in paragraph 3.1.2 we can exert some control over the accuracy of the finite element solution to some variational problem by changing the vertex density of our mesh. There are a number of tools which can generate meshes in such a fashion, that a prescribed density is met. Hence, a viable approach to a posteriori error estimation and adaptive grid generation consists in employing a suitable tool to generate some mesh, producing a numerical solution on that mesh, computing an improved vertex density with the aid of an a posteriori error estimator and in finally generating a new mesh. The above cycle may be repeated several times, until a satisfactory result

is found. However, since the meshes can be completely unrelated apart from the possibility that newer meshes may contain all the vertices already present in the older ones, the above approach has some serious drawbacks: The numerical data from the most recent computation has to be inserted into the new mesh, before it can provide e. g. starting values for a fixed point iteration. More sophisticated algorithms for the solution of the discretised variational problems, as they will be discussed in section 4.3, require hierarchies of meshes. As such hierarchies are usually not available, if an external tool has been employed for the mesh generation, they have to be created in an artificial manner, for instance by successive agglomeration of neighbouring elements (see e. g. [87] for details on such a procedure). Neither of these drawbacks is unsurmountable: However, the technical effort necessary to overcome them may become prohibitively expensive.

One possible alternative to the usage of density driven meshing procedures consists in the utilisation of *hierarchical meshes*. Such meshes sacrifice some flexibility in the layout of their topology but allow for a much improved handling of dynamically adapted mesh refinements. The basic idea behind hierarchical meshes consists in effecting the local grid refinement by the subdivision of single elements in a premeditated pattern, such that the resulting sequence of elements can be represented in a tree structure. Any "visible" element within the mesh belongs either to the primary grid, as it has been generated by some external tool, or has emerged in the process of grid refinement from exactly one other element, which has become "invisible" in the procedure. Speaking in terms of trees, we may assign to an element a root and a number of leaves: The root designates the very element, whence our current element has originated, while the leaves correspond to those elements which have emerged from our element by a subdivision. To help us fix our ideas we may envisage the mesh as a pile of bricks, such that the refined elements or bricks are buried beneath layers of smaller bricks which have emerged in the process of mesh refinement (as illustrated in the figure 4.1). With each element in the mesh we may associate a refinement level which designates its distance to the bottom of the pile respectively the number of edges we must traverse to reach the head of the tree.

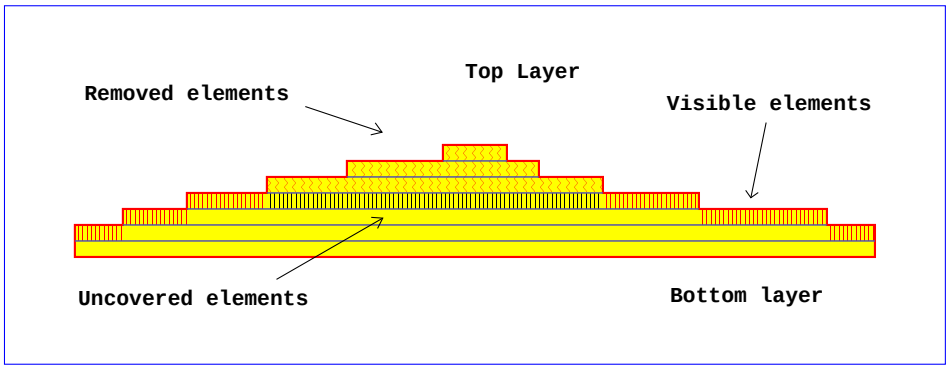
Figure 4.1: Hierarchical mesh refinement



Not all of the elements within the pile will be visited when we assemble the discrete variational problems: Only those visible from atop the pile will be accessed. Still we may not discard the invisible elements if the want to maintain the option of employing multilevel schemes to solve the discrete variational formulation. One possibility of supplying the hierarchy of meshes such schemes need consists in assembling the discrete problems on all those elements, which have become visible, after we have abraded a certain number of refinement levels from the top of the pile. By removing complete layers from the pile of elements one at a time we can construct the required hierarchy of discrete problems in one pass. The figure 4.2 indicates which elements are involved in such a procedure: to facilitate the assembly we obviously need to traverse the mesh in a horizontal fashion. In principle, we can implement a function that loops through all these elements using only the information from the tree structure and two auxiliary arrays, one of which contains pointers to all the head elements in the bottom layer, while the others serves as a stack. However, it is much simpler to store element handles in dedicated arrays level by level or to sort the elements in one large array as shown in figure 4.3.

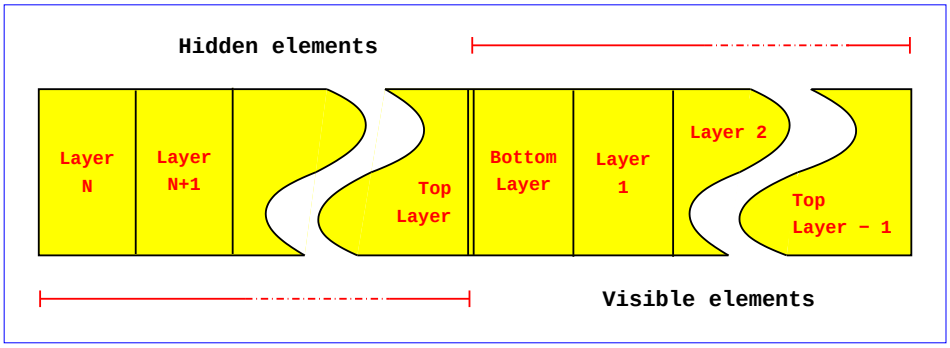
Using such an approach we arrange the mesh data simultaneously in a vertical and a horizontal

Figure 4.2: Assembly of finite element matrices



fashion. For technical reasons that will become apparent in the following two subsections, we need to furnish the mesh with yet another data structure: Let us equip each finite element with an array, which holds pointers to any neighbouring elements sharing the same refinement level. In case, certain such elements are nonexistent, a `NULL` pointer may be used instead. In consequence, the topology of each layer of the mesh is mapped onto a graph whose nodes are the finite elements and whose edges are represented by said pointers. In fact, each graph can be identified with the so called *Voronoi diagram* of its associated refinement level.

Figure 4.3: Storage layout for a hierarchical mesh



Since we are unable to warrant, that all finite elements in a mesh have the same number of neighbours, the same number of vertices and the same number of leaves, once they have been subjected to some refinement procedure, the above mentioned data structures need to be allocated dynamically. However, the allocation of storage on the heap requires a certain amount of additional space to be used by the operating system for accounting purposes. If large meshes must be manipulated, it therefore can prove more efficient in terms of memory consumption to implement a system of larger buffers which hold the usually short arrays of handles the elements are equipped with. Such a system will feature custom functions to handle the garbage collection.

Besides the topological data we have already described we need to store some additional information in each element: We will have to discriminate finite elements according to their type and to their location within the mesh. The former requirement is necessary to allocate dynamic memory appropriately. The latter requirement is necessary for example to distinguish various boundary conditions. In figure 4.4 the additional information has been moved into a base class termed "ElementParameter". Apart from a pointer to some data structure, which holds all the information, elements of the same type can share, said class must contain at least one material parameter, which can be assigned at will, an index, which the position of the element within the hierarchy of layers is referenced by (see figure 4.3), and a bit field to reflect the internal state of the element. With a view to reducing the memory footprint of the finite element software, the inclusion of an index is arguable. However, an index facilitates the easy removal of stale element

Figure 4.4: Possible definition of the class "FemElement"

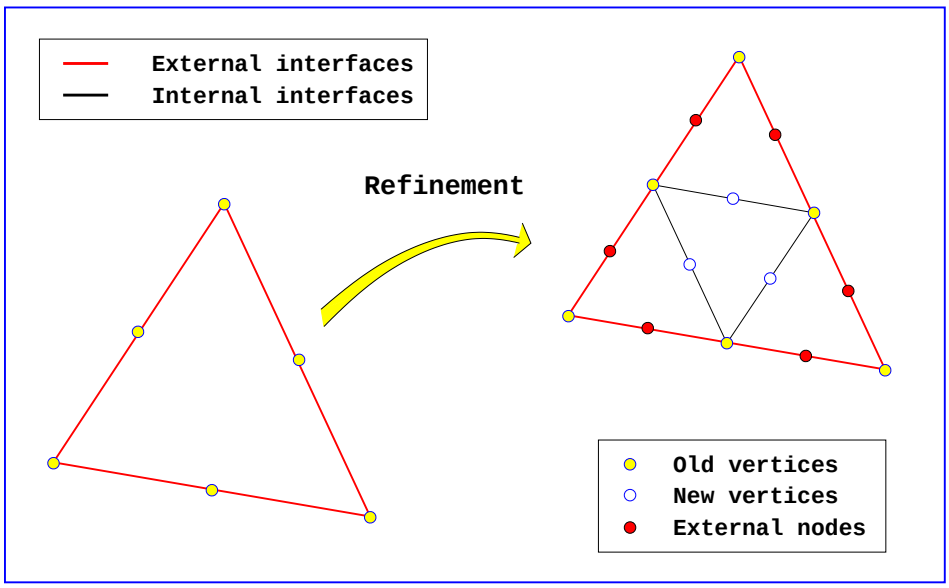
```
class FemElement :  public ElementParameter  {

    public:  ...

   private:  FemElement *Root ;
             FemElement **Leaf ;
             FemElement **Neighbour ;
             MeshNode **Vertex ;

} ;
```

handles from the level hierarchy. The expenditure of a bit field per element will be explained more fully in the sections 4.2.2 and 4.2.4.

4.1.2 On an algorithm for dynamic mesh refinement

The hierarchical refinement of some finite element can proceed in quite a number of ways. Since we need to control certain geometric properties of the emerging elements in order to ensure, that the approximation properties of the finite element interpolation do not deteriorate (see section 3.1.2), the most convenient refinement strategy consists in splitting the elements to be refined into a number of self-similar shapes. In this process we have to create a certain number of new vertices, while others may be available due to neighbouring elements, which have already been split. If the vertices are contained in the interior of the root element the assignment of vertex data and topological information is trivial and may be defined for example in a configuration file, as only such elements are involved, which are created in the same instance. Those vertices which are located on the interfaces of the root element are more difficult to deal with.

Figure 4.5: Regular refinement of an isoparametric P2-element



If one of the neighbouring element has been split previously, the vertices on all of the common interface must be identified and pointers to these vertices registered with the newly formed leaf elements. If there are no adjacent elements on the same refinement level, new vertices have to

be allocated. The topological information has to be updated accordingly. However, for technical reasons we will address in section 4.1.3 we cannot view the process of mesh refinement as a purely local manipulation: Adjacent elements must never have refinement levels that differ by more than one level. In figure 4.1 a violation of the above condition is depicted, termed "level gap". Consequently, the splitting of a single element may be preceded by an avalanche of element refinements taking place on lower layers.

Figure 4.6: Outline of refinement algorithm

```
Algorithm SplitElement.

  Input:  Mesh *M0, FemElement *E0, Refinement Level L

 1.)  If ( E0->Leaf != NULL ) terminate.
 2.)  Determine split scheme, initialise handle *S ;
 3.)  Set l=0. While ( l++ < S->NumberOfLeaves ) do:
      1.)  Find description of leaf l, initialise handle *DL ;
      2.)  E0->Leaf[l] = M0->AllocateNewElement( DL, L ) ;
      3.)  E0->Leaf[l]->Root = E0 ;
      4.)  E0->Leaf[l]->Leaf = NULL ;
      5.)  Copy handles to old vertices ;
      6.)  Allocate new vertices ;
 4.)  If ( E0->Root == NULL ) go to 8 ;
 5.)  Determine index i with Root->Leaf[i] == E0 ;
 6.)  Find description of leaf i, initialise handle *D0 ;
 7.)  Set n=0. While ( n++ < D0->NumberOfExternalInterfaces ) do:
      1.)  Determine index k of interface number n ;
      2.)  Call SplitElement( M0, Root->Neighbour[k], L-1 ) ;
 8.)  Set l=0. While ( l++ < S->NumberOfLeaves ) do:
      1.)  Find description of leaf l, initialise handle *DL ;
      2.)  Set n=0. While ( n++ < DL->NumberOfExternalNodes ) do:
           1.)  Fix handle *EN for adjacent element on level L-1 ;
           2.)  If ( EN->Leaf == NULL )
                  Allocate a new vertex, initialise handle *V ;
                else
                  Recover *V from among EN->Leaf[*]->Vertex ;
           3.)  E0->Leaf[l]->Vertex[DL->NodeIndex[n]] = V ;
 9.)  Set l=0. While ( l < S->NumberOfLeaves ) do:
      1.)  Find description of leaf l, initialise handle *DL ;
      2.)  Set n=0. While ( n++ < DL->NumberOfSharedNodes ) do:
           1.)  Fix destination k of vertex number n ;
           2.)  Fix source index j of shared vertex ;
           3.)  Determine index i of associated leaf ;
           4.)  E0->Leaf[l]->Vertex[k] = E0->Leaf[i]->Vertex[j] ;
10.)  Update topological data of the mesh *M0 ;
11.)  M0->UpdateElementArray( E0, L ) ;
```

The figure 4.6 is meant to summarise the ideas we have presented so far. Though the notation bears a slight resemblance with the programming language *C*, the figure is not intended to provide actual code. Implicitly, we have introduced a number of data structures, which have not yet been commented on. However, it should be obvious by our nomenclature what purposes these data structures serve. They contain basically a number of index tables, that encode all the necessary topological information. The tables are initialised prior to the manipulation of any grids with the help of a parser, whose grammar is detailed in the appendix A.

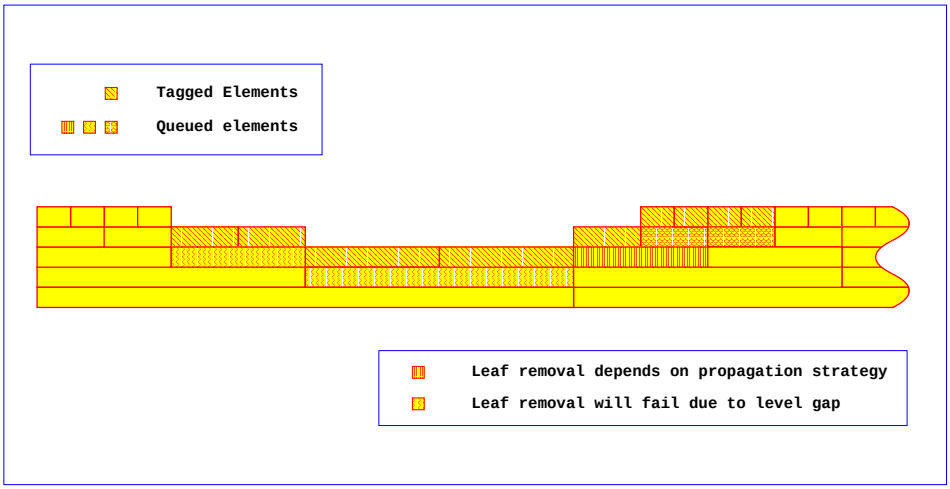
Remark 4.1 In principle it is possible to deploy the refinement algorithm we have outlined in figure 4.6 every time a local mesh adaption has been found expedient. As the refinement can affect larger areas of the mesh we must eventually discard some part of the algebraic data already

assembled on the unrefined mesh and recompute the missing data on the newly created elements, whenever the refinement routine has been called. Such a procedure requires a close interaction between the application specific code to be supplied by a user and the mesh handling routines as they are collected in the finite element library. Hence, its implementation would conflict with a clean design of the libraries high-level interface. A further complication lies in the fact, that the refinement procedure is usually controlled by assessing the numerical solution on an element by element basis. Therefore, the numerical solution should be updated, any time the discretisation changes. However, the computational effort necessary for such continuous updates may be substantial or even prohibitive, depending on the underlying variational problem. In consequence, it seems most apposite to only mark those elements which need to be refined and defer the actual mesh update until all elements have been inspected. The elements may either be tagged, using their internal bit field for that purpose, or stored in a dedicated list.

4.1.3 On the removal of elements from a mesh

While we can split any visible element in a mesh regardless of its adjoining elements, we may not simply remove a cluster of elements who happen to be leaves of the same element and put that common root in their place. With a view to avoiding level gaps we have to make sure, the replaced elements are surrounded by elements of at most the same refinement level only.

Figure 4.7: Cascaded removal of leaf elements



Though the removal of leaf elements is a purely local procedure, the necessity of monitoring the neighbourhood for possible level gaps prevents us from treating all the elements to be removed independently of each other. Specifically, we must account for the contingency, that elements of a higher refinement level prevent the coarsening of the mesh, which are themselves scheduled for removal. A simple possibility of avoiding unnecessary obstructions consists in queuing those elements, whose leaves have been marked for deletion, and updating the mesh subsequently in one pass. Thereby, we start with those elements, whose refinement level is the highest, and proceed to elements with lower refinement levels in a layer by layer fashion. In this cascaded procedure we have two options how we want to deal with clusters of marked elements: After we have successfully removed the leaf elements we can tag their root to indicate, that it is eligible for removal as well, once the next lower mesh layer will be processed. Alternatively, we may desist from propagating the tag. Employing the first strategy we can ensure, that extensive areas of the mesh containing elements with various levels of refinement can be subjected to the mesh coarsening at once. There is a serious drawback however: If we tag the top layer of the mesh, merely the bottom layer will remain after a mesh update. Since such a behaviour is in stark contrast to the performance of the refinement algorithm as outlined in figure 4.6, which can never add more than one refinement level to the mesh at a time, we prefer the second, more conservative strategy. By disabling the propagation of the tag we ensure, that no more than one refinement level can be removed at once. We can no longer warrant, on the other hand, that the

majority of the elements we have marked for deletion will be actually removed in a mesh update. The situation is illustrated in figure 4.7.

Figure 4.8: Outline of mesh coarsening algorithm

```
Algorithm RemoveLeaves.

  Input:  Mesh *M0, FemElement *R0, Refinement Level L

1.)  Determine split scheme, initialise handle *S ;
2.)  Set l=0. While ( l++ < S->NumberOfLeaves ) do:
     1.)  If ( E0->Leaf[l]->Leaf != NULL )  terminate.
     2.)  Test removal tag of E0->Leaf[l] ;
     3.)  If the tag is missing, terminate.
3.)  Find description of element *R0, initialise handle *D0 ;
4.)  Set n=0. While ( n++ < D0->NumberOfNeighbours ) do:
     1.)  Find split scheme of R0->Neighbour[n], fix handle *SN ;
     2.)  Fix index i with R0->Neighbour[n]->Neighbour[i] = R0 ;
     3.)  Fix interface description EX = SN->InterfaceData[i] ;
     4.)  Set l=0. While ( l++ < EX->NumberOfElements ) do:
          1.)  Determine index j of adjacant leaf number l ;
          2.)  EL = R0->Neighbour[n]->Leaf[j] ;
          3.)  If ( EL->Leaf != NULL )  terminate.
5.)  Set l=0. While ( l++ < S->NumberOfLeaves ) do:
     1.)  Find description of leaf l, initialise handle *DL ;
     2.)  Set n=0. While ( n++ < DL->NumberOfExternalNodes ) do:
          1.)  Determine index j of associated interface ;
          2.)  EN = R0->Leaf[l]->Neighbour[j] ;
          3.)  If ( EN == NULL )  Free vertex data ;
          4.)  Free data of internal and shared vertices ;
          5.)  M0->RemoveElement( R0->Leaf[l], L+1 ) ;
          6.)  Update topological data of element *EN ;
6.)  R0->Leaf = NULL.
```

To schedule elements for removal the following procedure may be adopted: For each layer of the mesh a separate list of element handles is maintained. One bit of each element's internal bit field is assigned to indicating, if the element or one of its leaves is going to be deleted from the mesh. Every time a top layer element is marked for removal, this tag is set. If the root element hasn't been tagged before, it is also marked for removal. In addition, a pointer to the root element is appended to the proper list as indicated by the level index of the root element. Thus, the storage of duplicate handles is prevented. Eventually, the mesh update is invoked and all elements marked for refinement are split. Thereafter, the lists of element pointers are traversed in turn starting with the top layer. For each element in these lists the algorithm described in figure 4.8 is deployed. Their tags and those of their leaves are cleared, as far as they have been accessory to the mesh adaption.

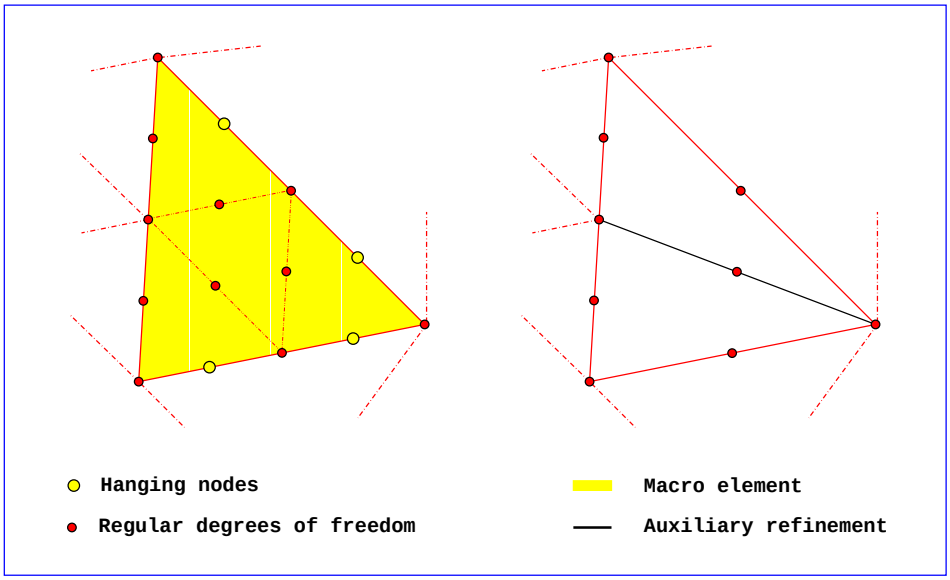
4.1.4 *On the use of auxiliary refinement schemes*

The generation of locally adapted meshes with the help of hierarchical refinement techniques has been discussed in some depth in the two preceding subsections. We have outlined an algorithm, by which we can introduce new elements of smaller size into a given mesh, and have discussed a method to agglomerate a cluster of elements and to replace them with their common root. Our deliberations have been confined, however, on the geometrical aspects of the adaption procedures. Once the mesh has been generated, the degrees of freedom must be identified, by which the numerical solution of our variational problem will be described. In 3.1.1 we have outlined the general principles which the representation of the numerical solution revolves around: In order to supply globally defined ansatz spaces the local degrees of freedom are to be coupled by algebraic

constraints of the form (3.3). If we employ a conforming discretisation scheme of Lagrange type on a regular mesh, these algebraic constraints are usually of a very simple form with $B_{li}^{j} \in \{-1, 0, 1\}$. Consequently, one portion of the linear factors can be eliminated in favour of the remaining faction in a transparent fashion. We may argue, that we compute a Schur complement in assembling the algebraic formulation of our variational problem, even though the supernumerary local degrees of freedom are suppressed entirely.

The situation is changed fundamentally, when we introduce local mesh adjustments. Since the mesh width can now change from one element to an adjoining element, the algebraic constraints (3.3) are no longer that simplistic as they may present themselves in the case of uniformly refined meshes. Basically, we have three options how to treat those regions which interface elements of a different refinement level: We can compute a Schur complement just as we would have done for a uniformly refined mesh. While such a computation would require merely a proper assignment of indices to the local degrees of freedom in the case of an uniformly refined mesh, we must actually carry out a static condensation to eliminate certain linear factors from the algebraic problem, if the mesh refinement is nonuniform. The set of algebraic constraints is thereby enriched by additional conditions either to ensure a certain regularity of the global shape functions or to warrant certain approximation properties of these shape functions on the element faces. The second option consists in employing a hybrid discretisation scheme: for we can match elements of different refinement level without deteriorating the global approximation properties of the ansatz, if we choose the Lagrange multipliers on the elements interfaces judiciously. Depending on our choice of these multipliers the hybrid method may be equivalent to the first approach. As a third option we may avoid interfacing elements of different refinement level altogether. We can achieve this goal with the help of auxiliary refinement patterns, which we deploy in addition to the regular mesh refinement in compliance with the geometrical properties of the mesh.

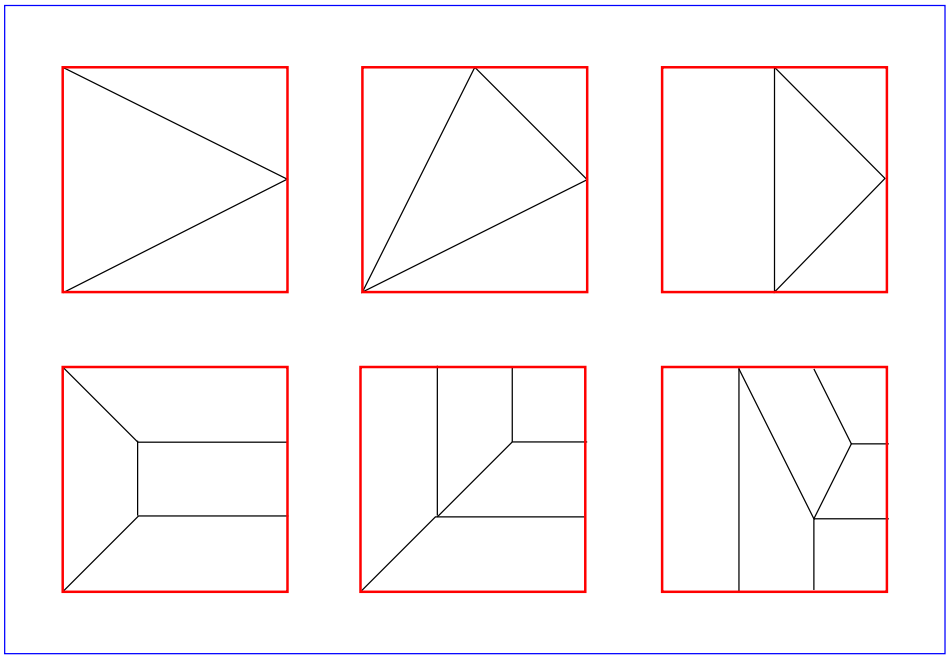
Figure 4.9: P2 Macro-elements versus auxiliary refinements



Pursuing the first option we are faced in almost all cases with the necessity of eliminating superfluous degrees of freedom related to those element interfaces which join elements of different refinement level. As the most popular finite elements of Lagrange type allow to identify their degrees of freedom with the vertices of the mesh, these supernumerary degrees of freedom are usually referred to as *hanging nodes* . The transparent treatment of elements that feature hanging nodes is not entirely trivial. One possibility consists in the implementation of a filter, that enforces the algebraic constraints (3.3) after every matrix vector multiplication. Once such a filter is in effect, we can assemble local problems for each element of the mesh without regard for of any hanging nodes. The simplicity of the approach is counterbalanced by a severe drawback, however: Since the assembly of a global problem is impossible, sophisticated numerical techniques cannot be applied to solve the algebraic formulation. Alternatively, we may introduce special macro

elements as indicated in figure 4.9. These elements have a variable number of degrees of freedom and feature special shape functions to account for any requirements to be imposed on the global ansatz. Cubature formulae may be obtained by applying an ordinary cubature scheme to each of the constitutive leaf elements.

Figure 4.10: Auxiliary refinement patterns for the Q1 element



As long as we employ parametric elements only, the construction of suitable macro elements can be done on the reference patch. The bulk of the data involved in the element setup can be calculated beforehand and retrieved from tables when necessary. Thus, we can ensure that the computational overhead of creating element macros stays within reasonable limits. A number of important finite element discretisations, for instance, those based on elements of Raviart-Thomas type (see [122] for a description), do not lend themselves to such a treatment, however. A minor drawback of using macro-elements lies in the necessity of querying both the degrees of freedom and the number of cubature nodes, before a local problem can be assembled: due to the ever changing nature of the elements and their properties no preconcerted shortcuts can be exploited safely to accelerate the assemblage of the discrete variational formulation.

Hence, the only possibility of providing a finite element library with a consistent and safe application interface consists in implementing auxiliary refinement patterns, that "catch" the hanging nodes. Such refinements are applied after the regular mesh transformations have been carried out, as they have been discussed in the previous paragraphs. Conversely, before the regular mesh adaption can take place, the special refinement patterns must be removed. Both the installation and the removal of the auxiliary refinements can be done basically in the same manner we have described in the figures 4.6 and 4.8.

However, depending on the nature of the hierarchical refinement it can be difficult to supply suitable refinement patterns for all geometrical configurations possible. In such situations the incriminated element may be split in a conventional way. Alternatively, elements of a different type may be employed, which fit into the configuration. For instance, Courant triangles may be used in auxiliary refinement patterns for the rectangular finite element with continuous, piecewise bilinear shape functions, usually referred to as Q1 element. When missing auxiliary patterns are replaced by standard refinements, the mesh width may shrink in an area, that is somewhat larger than intended. Since the standard refinement is permanent, this could pose a problem in some situations. One possibility of avoiding avalanche effects consists in the use of more complex hierarchical refinement schemes, which can cover a broader range of geometrical configurations. In figure 4.10 some examples are sketched.

4.2 Merging and matching of meshes

In the previous paragraphs we have focused on the technical prerequisites for dynamic mesh adaption. In the following we shall enlarge on an application of locally refined meshes, which is not as obvious as the control of some approximation error by judicious mesh transformations: we will discuss a method of merging two meshes, such that the resulting mesh can be used to construct an ansatz, which contains both of the original trial spaces. By merging two meshes we will understand the construction of a mesh, whose geometrical complexity is minimal, while it still contains all the vertices and all the edges, that are present in both of the original meshes. A merging algorithm of that description is useful, whenever data that has been obtained on one mesh needs to be transferred onto another mesh. Since we may also employ shape functions of different make on the original meshes, their joint "super-mesh" can further be deployed to calculate expressions, that involve numerical approximations obtained by distinct finite element schemes. Especially, we can exploit the technology to evaluate those duality based a posteriori error estimates, we have discussed in section 3.3.

In the first paragraph of this section we will digress on a technical device that proves useful, whenever certain entities of one or more meshes must be correlated: the definition of artificial order relations which enable us to solve the matching problem by a standard sorting procedure. The next paragraph will also cover a somewhat subsidiary problem: how to keep track of auxiliary refinement patterns that have been removed in the process of subsequent mesh refinements. Both of these problems must be addressed in order to keep the numerical complexity of the grid merge within acceptable limits. In the third paragraph we will present the merging algorithm, we have actually implemented in our finite element library. The selection of proper refinement patterns, which concludes the construction of the compounded mesh, will be outlined in section 4.2.4.

4.2.1 Identifying mesh entities by Sorting

The basic idea underlying the merging algorithm, we will examine in the next but one section, consists in establishing a one to one correspondence between those elements, which constitute the bottom layer of the one mesh, and those elements, which form the bottom layer of the other mesh. Once such a mutual relationship has been established, each element in either of the meshes can be uniquely identified by the path, along which we must traverse the level hierarchy in order to reach the bottom layer, and the very element within the bottom layer, that is the endpoint of said path. The matching of two elements which belong to distinct meshes may now proceed in the following fashion: Starting from one element a path reaching towards the bottom layer of the mesh is constructed. On each refinement level, the index of the recently visited leaf element is pushed on a stack, such that the path can be retraced. When the bottom layer is reached, the corresponding elements in the respective meshes exchange their roles and the path is backtracked.

The above procedure can only work, if both meshes are obtained by hierarchical refinement and if their respective bottom layers share the same geometry. However, as these meshes must resolve the same computational domain, neither of the above assumptions is unduly restrictive. To substantiate our notions, let us suppose that both the mesh for the primal formulation of some variational problem and the mesh for its dual formulation have been derived from the very same topological data provided by some external mesh generator. Unfortunately, we may not infer from this plausible premise, that our two meshes will store the element information for the bottom layer in exactly the same layout. Hence, we need to rearrange the data of each bottom layer in an unique way, which is based on the geometrical properties of their respective elements alone. The naive approach to establishing the correspondence between the elements of both layers consists in comparing each element in one layer against each element in the other layer. The numerical complexity of such a procedure is given by N^2, if N denotes the number of elements in each bottom layer. Since the bottom layers may already contain a substantial number of elements, we clearly need to find more sophisticated algorithms to organise the data.

One approach to rearranging the elements according to certain geometrical criteria relies on the definition of an order relation on the set of all elements which constitute the mesh. Obviously, such an order relation must be total. Preliminary to the construction of the order relation let us introduce a relation, which compares vectors from the space $\mathbb{R}^n$:

$$\{x, x_{n-1}\} \prec \{y, y_{n-1}\} \quad :\Longleftrightarrow \quad x < y \quad \vee \quad (x = y \ \wedge \ x_{n-1} \prec y_{n-1}) \ . \tag{4.1}$$

In the case $n = 1$ the second condition is void. When neither $x \prec y$ nor $y \prec x$ hold for two vectors $x, y \in \mathbb{R}^m$, we find that x equals y. If we want to compare two parametric elements $A, B \in \mathcal{M}_h$, the above relation may be applied for instance to their centres of gravity. With a view to definition 3.1 we can describe the geometry of these elements by specifying the set of points $\{a_1, \dots a_K\} \subset \mathbb{R}^n$ respectively $\{b_1, \dots b_K\} \subset \mathbb{R}^n$ which define the transformation (3.7). Hence, one possible order relation can be introduced by:

$$A \prec B \quad :\Longleftrightarrow \quad \sum_{k=1}^{K} a_k \prec \sum_{k=1}^{K} b_k \quad . \tag{4.2}$$

The advantage of using the order relation (4.2) lies in its definition being independent of the particular sequence, in which the nodes $\{a_1, \dots a_K\}$ and $\{b_1, \dots b_K\}$ are arranged. There is a drawback with the above solution, however. The centre of gravity need not necessarily be contained within the element, if we allow for elements of lower dimension to represent interfaces. In applications which contain curved element boundaries the above order relation may therefore cause ambiguities we would have to detect and resolve in a post-process.

Alternatively, we may collect the components of the vertices $\{a_1, \dots a_K\}$ and $\{b_1, \dots b_K\}$ in two larger vectors $\mathfrak{a}, \mathfrak{b} \in \mathbb{R}^{K \times n}$ and eventually compare these. Since such vectors depend on the enumeration of the vertices as well as on their coordinates, let us introduce a permutation $\pi : \{1, \dots K\} \longmapsto \{1, \dots K\}$ with the property $a_{\pi(i)} \prec a_{\pi(i+1)}$ and define:

$$\mathfrak{a} \quad := \quad \left\{ a^{(1)}_{\pi(1)}, \dots a^{(n)}_{\pi(1)}, a^{(1)}_{\pi(2)}, \dots a^{(n)}_{\pi(2)}, \dots a^{(1)}_{\pi(K)}, \dots a^{(n)}_{\pi(K)} \right\} \quad .$$

The vector $\mathfrak{b}$ is constructed in an analogous fashion. Since there are no elements with collapsed nodes in a regular mesh, the permutation π is well defined. The vectors $\mathfrak{a}$ and $\mathfrak{b}$ uniquely describe their respective elements: the identity $\mathfrak{a} = \mathfrak{b}$ implies that A and B are spanned by the very same vertices. We infer, that we may replace the relation (4.2) by:

$$A \prec B \quad :\Longleftrightarrow \quad \mathfrak{a} \prec \mathfrak{b} \quad . \tag{4.3}$$

With a view to the geometrical properties of the mesh we have to decide which of the order relations (4.2) or (4.3) we want to use. If the element mappings (3.7) are affine, the former relation should clearly be preferred, as it is much simpler to implement and faster to evaluate. The latter relation may be the superior choice, if we must handle complex geometries and distorted elements. However, if the meshes feature strong local refinements a straightforward implementation of (4.1) may fail to properly identify corresponding elements regardless which of the discussed relations we employ. Such a failure would be due to rounding errors, which have caused a slight shift of the vertex coordinates. A solution consists in the following modification to (4.1):

$$\{x, x_{n-1}\} \prec_\varepsilon \{y, y_{n-1}\} \quad :\Longleftrightarrow \quad x < y - \varepsilon \quad \vee \quad \left(x < y + \varepsilon \quad \wedge \quad x_{n-1} \prec_\varepsilon y_{n-1} \right)$$

whereby $\varepsilon > 0$ denotes a sufficiently small parameter. In effect, cubes with a volume of about ε^n are substituted for the points we have considered above. If neither $x \prec_\varepsilon y$ nor $y \prec_\varepsilon x$ holds, we can therefore no longer conclude $x = y$. Instead, we infer from the above definition $\|x - y\|_\infty < 2\,\varepsilon$. Naturally, our choice of the parameter ε must depend on the local mesh size and may vary from one region of the computational domain to another.

Remark 4.2 Once a suitable order relation on the set of those mesh entities has been defined, which we want to correlate with their counterparts from a second mesh, we can apply basically any sorting algorithm to both data sets. Afterwards, corresponding mesh entities can be identified by traversing both sets simultaneously. Clearly, the numerical complexity of the sorting algorithm should be as low as possible: however, the performance of such algorithms is usually determined under certain assumptions on the distribution of the keys we want to sort. For instance, we may not use the famous *Quick Sort* algorithm [84, 90] despite its average numerical complexity of $\mathcal{O}(N)$, as the complexity deteriorates to $\mathcal{O}(N^2)$ on ordered sets. If the data must be rearranged in situ, the *Heap Sort* algorithm [90, 130] is a good choice, since its average and its worst case complexity are the same: $\mathcal{O}(N \log N)$. If a link field is available or must be allocated in any case to prevent excessive data movement, the *Merge Sort* algorithm [90] is even superior, as it requires significantly less comparisons than the heap sort procedure. A particularly elegant implementation of this latter algorithm can be found in [38].

Remark 4.3 The above deliberations have focused on the identification of elements by sorting them in accordance with geometrical criteria to be met by their vertices. However, the concepts we have presented may also be employed in identifying the global degrees of freedom which are associated with an ansatz $\mathfrak{V}$. To this end we must define an order relation on the set $\mathcal{S}$ of local degrees of freedom. If the finite elements are of Lagrange type, each functional $\sigma \in \mathcal{S}$ corresponds to an unique point $x \in \Omega$ of the computational domain. Hence, we can easily define the required order relation defining by comparing the corresponding points:

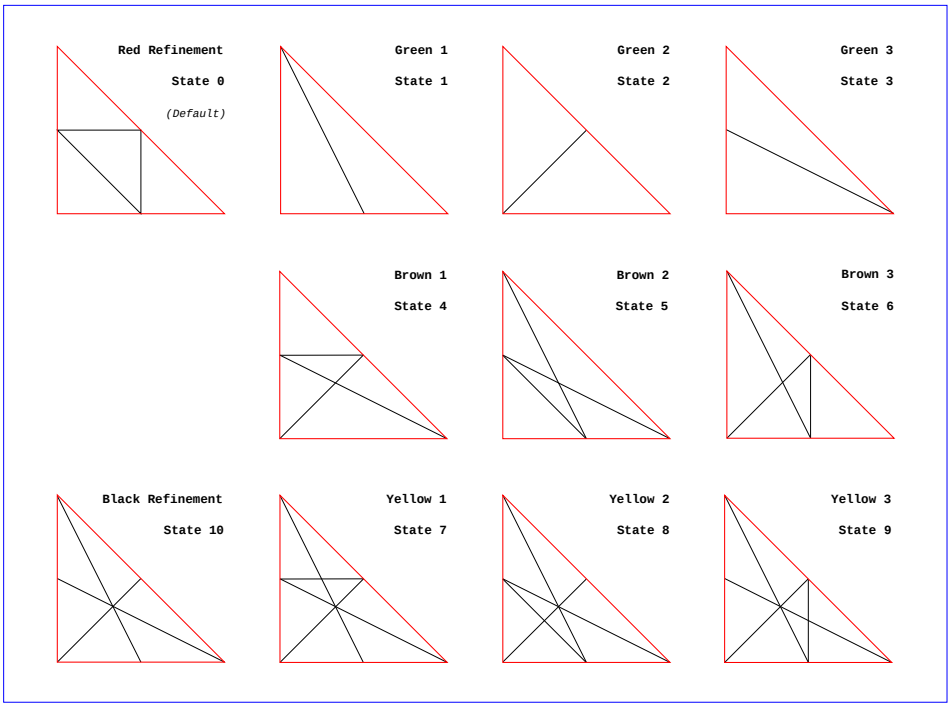
$$\sigma \prec \hat{\sigma} \qquad :\Longleftrightarrow \qquad x \prec_{\varepsilon} \hat{x} \quad .$$

If the finite elements are of a different kind, we are usually still able to associate some geometrical data with a functional $\sigma \in \mathcal{S}$ if we supplement this information with a key. To give an example: We associate the first moment of the normal flux, which serves as a local degree of freedom for the Raviart-Thomas element [122], with the middle point of the edge, across which we want to approximate the flux. In addition, we assign the key "0" to this degree of freedom. Any moments of higher order we associate with the very same coordinates. However, these moments receive keys consecutively numbered. The key may be treated like an additional coordinate axis such that the recursive definition 4.1 can be applied.

4.2.2 Management of auxiliary Refinement Patterns

In the following we will assume, that we have generated with the help of some hierarchical refinement procedure two meshes $\mathcal{M}_{h'}^{(1)}$ and $\mathcal{M}_{h''}^{(2)}$ that share the same bottom layer. Our aim is the construction of a new mesh $\mathcal{M}_h$ containing all the edges and all the vertices that are present either in $\mathcal{M}_{h'}^{(1)}$ or $\mathcal{M}_{h''}^{(2)}$. Since both of our meshes may be locally refined, we must account hereby not only for the regular refinement patterns, but for the auxiliary patterns as well.

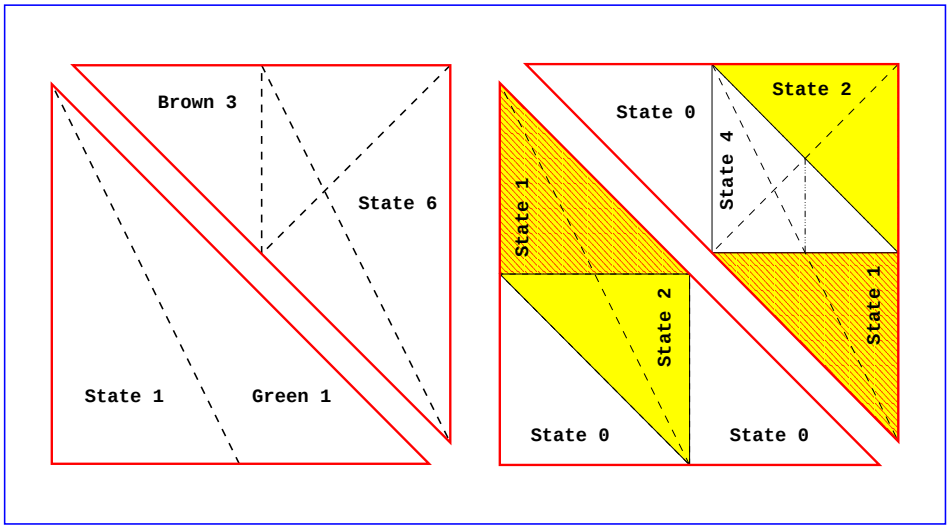
Figure 4.11: Refinement patterns for triangular elements



The treatment of any conventionally refined elements is simple, since all regular subdivisions are of a permanent nature: as far as only they are concerned, the topology of the joint mesh $\mathcal{M}_h$ can be recovered directly from the tree structure of both $\mathcal{M}_{h'}^{(1)}$ and $\mathcal{M}_{h''}^{(2)}$. In figure 4.14 we outline our implementation of a depth first strategy to augment the bulk of a mesh, which consists of regularly refined elements, with those elements constituting the bulk of a second mesh. The resulting mesh contains all the vertices and all the edges, that would be present in either of both

meshes, if we ignored special refinement patterns. Let us suppose, the method `PrepareMerge` is called for the mesh $\mathcal{M}_{h'}^{(1)}$ while $\mathcal{M}_{h''}^{(2)}$ is the first argument of the parameter list. The second argument is a handle for some sort of hash table, which encodes the correlation between the elements in the bottom layer of $\mathcal{M}_{h'}^{(1)}$ and those elements in the bottom layer of $\mathcal{M}_{h''}^{(2)}$. We compile this table externally, since it will also be used in the merging algorithm itself. We note, that any auxiliary refinement patterns, which may be present on the surface, so to speak, of the mesh $\mathcal{M}_{h'}^{(1)}$ are removed, before any new elements are created. In the process, additional edges and perhaps even vertices may be lost, that must be restored when the special refinement patterns are placed on the "surface" of the joint mesh $\mathcal{M}_h$.

Figure 4.12: Propagation of Refinement States

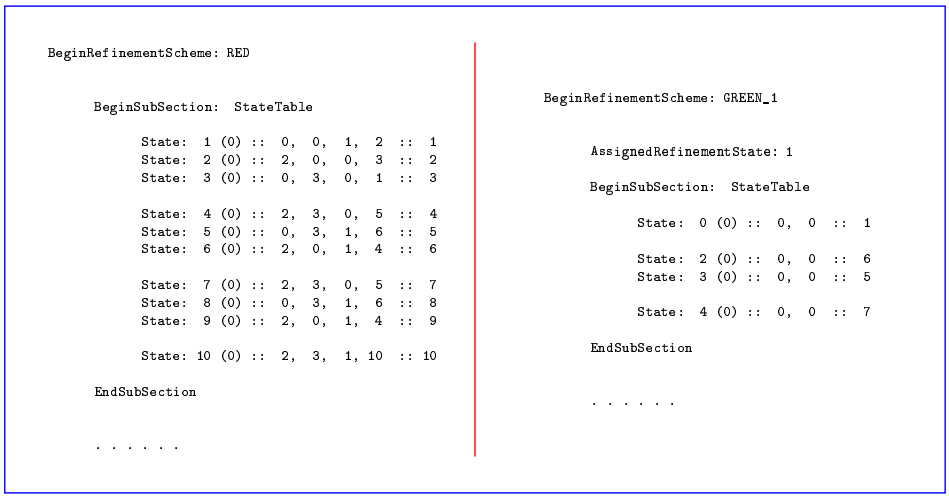


We conclude, that we must preserve the information about the auxiliary refinement patterns, even though these patterns themselves are cleared away. The simplest possibility of storing the information consists in assigning to each element a certain state, if an auxiliary refinement pattern on top of it has been removed. To hold said state the bit field is employed, with which we have equipped each element (see section 4.1.1). Subsequently, more and more elements are added to the mesh $\mathcal{M}_{h'}^{(1)}$, until its bulk is a superset of those elements, which constitute the bulk of $\mathcal{M}_{h''}^{(2)}$. Whenever elements are refined in a regular fashion, whose state engine signals the former presence of additional edges, the information must be distributed among the newly created leaf elements. Depending on the auxiliary refinement pattern, which has been removed, the leaves will be affected in different ways. Hence, we may not simply propagate the internal state of the root element to its children. In figure 4.11 all the refinement pattern are depicted, which we must employ in the construction of the new mesh $\mathcal{M}_h$. To simplify the referencing, we have given names to these patterns: The denominations "red" and "green" are widely known and need not be explained. The "brown" patterns may be used, whenever an element with a nonzero state has to be split in order to avoid hanging nodes on one of its interfaces. If several meshes must be merged, we have to account for brown patterns as well as for green ones. In situations like that, the "yellow" patterns and the "black" one can be employed to recover any edges and nodes, which stem from brown refinements discarded in the very first step of the method `PrepareMerge`.

In figure 4.12 we give two examples, how the refinement state of an element is propagated to its leaves when the element is split in a regular fashion. Between themselves, the three green, the three brown and the three yellow refinement patterns do not really differ from a geometrical point of view. We have introduced them nevertheless to encode not only the general form of the refinement but also its alignment with respect to the edges of the root element. Thus, the orientation of any special refinement pattern to be arranged on the surface of $\mathcal{M}_h$ can be inferred immediately from the state engine of the corresponding root element.

Figure 4.13 illustrates how state transitions are specified in our finite element description language. As the grammar of this language is detailed in the appendix A let us give but a few remarks on the information provided in the two examples. Assuming some finite element is

Figure 4.13: Extracts from element description files



about to be split according to the refinement scheme "red", four leaf elements will be generated. Their state will depend on the present state of their future root element. Each line of the two tables in figure 4.13 corresponds to a possible state of the future root element. The number in brackets refers to the state of the mesh and allows the implementation of several refinement strategies which we switch between by changing the mesh state. In our two examples only the default state is used. Following a field separator there is the colon separated list of states which the leaves must assume. Each list is terminated by another field separator and a last number. This number designates the very state, the root element must assume, once its leaves have been removed again. As the green refinement has an auxiliary character, the leaf elements are cleared away before the regular mesh refinement is carried out. Hence, the leaf elements themselves are never refined. Accordingly, the state data for leaves, which is critical in the case of the regular "red" scheme as depicted on the left, is irrelevant for the auxiliary "green" scheme described on the right hand side of figure 4.13. In our example, these states have been set to their default value. When a regular refinement is reversed, the state of the former root element is not affected. When an auxiliary refinement pattern is removed, however, the last number in each row provides the very mechanism to keep track of those additional nodes and edges, that are destroyed when the method `FemGrid::RemoveAuxiliaryRefinement` is invoked.

4.2.3 The Description of a Merging Algorithm

In the previous section we have discussed how auxiliary refinement patterns, that have been removed prior to mesh refinements, can be recovered with the help of finite automata. Loosely speaking, we can define suitable state transitions and thus invest each element in our mesh with a memory for those edges and nodes we have removed along with the auxiliary refinement patterns. Unfortunately, our device is not powerful enough to facilitate the merging of two locally refined meshes: While one mesh $\mathcal{M}_{h'}^{(1)}$ may feature a strong local mesh refinement around some point $x_1 \in \Omega$ of the computational domain, the other mesh $\mathcal{M}_{h''}^{(2)}$ may be locally refined around some distant point $x_2 \in \Omega$. Let us assume, the second mesh is practically unrefined around x_1 while $\mathcal{M}_{h'}^{(1)}$ is unrefined around the point x_2. If we invoke the method outlined in figure 4.14 for the mesh $\mathcal{M}_{h'}^{(1)}$ and use the other mesh as the first argument in the parameter list, we will not encounter significant difficulties in the immediate neighbourhood of the point x_2. The information about the auxiliary refinement we have removed from the top layer of $\mathcal{M}_{h'}^{(1)}$ is correctly propagated to the top layer of the joint mesh, as $\mathcal{M}_{h''}^{(2)}$ has a deeper hierarchy of levels around that point than $\mathcal{M}_{h'}^{(1)}$. In a neighbourhood of the point x_1 the roles of both meshes should be reversed. As that is not possible, the state engines of all those elements which belong to the joint mesh and lie close to the point x_1, are not updated at all.

Figure 4.14: Depth first construction of the mesh bulk

```
void FemGrid::PrepareMerge( FemGrid &G0, FemElement **TransferTable )
{
  unsigned Stack[ FEM_MAX_NUMBER_OF_LEVELS ] ;
  RemoveAuxiliaryRefinement() ;
  unsigned k = G0.LevelData[0] -> EndOfLayer ;
  while ( k )
  {
    FemElement *EZ = G0.ElementArray[ -- k ] ;
    if ( EZ -> Leaf != NULL )
    {
      if ( EZ -> TestAuxiliaryRefinement() )   continue ;
      FemElement *E0 = TransferTable[ EZ -> ElementIndex ] ;
      E0 -> Split() ;
      unsigned StackPointer( 0 ) ;
      unsigned n( EZ -> NumberOfLeaves() ) ;
      while ( EZ != NULL )
      {
        while ( n )
        {
          FemElement *EL = EZ -> Leaf[ -- n ] ;
          if ( EL -> Leaf != NULL )
          {
            if ( EL -> TestAuxiliaryRefinement() )  continue ;
            Stack[ ++ StackPointer ] = n ;
            E0 -> Leaf[ n ] -> Split() ;
            E0 = E0 -> Leaf[ n ] ;
            EZ = EL ;
            n = EZ -> NumberOfLeaves() ;
          }
        }
        E0 = E0 -> Root ;
        EZ = EZ -> Root ;
        n = Stack[ StackPointer -- ] ;
      }
    }
  }
}
```

We conclude, that we must still propagate any information about auxiliary refinements of the mesh $\mathcal{M}_{h''}^{(2)}$ to the top layer of the joint mesh $\mathcal{M}_h$ even though we have already invoked the routine presented in figure 4.14. The algorithm which updates the state engines must proceed through four stages: i) Any auxiliary refinement pattern on the surface of the mesh $\mathcal{M}_{h''}^{(2)}$ must be identified. ii) Once such a pattern has been detected, the counterpart of its supporting root element in the joint mesh must be found. How this can be done, has already been described, though briefly, in the beginning of section 4.2.1. iii) The proper refinement state of the newly found "root" in the joint mesh is determined. iv) Using a depth first strategy the state engines of all the elements on top of this pivot are updated. Figure 4.15 describes the basic layout of our implementation of the above procedure. To save space the second stage of the algorithm has been subsumed under the denomination "Section A", while its third stage has been abbreviated by "Section B". The depth first strategy for updating the element states is summarised as "Section C". In the following let us discuss these functional blocks separately.

Only those elements can form a part of a special refinement pattern, which are located on the surface of the hierarchy of mesh layers as it is illustrated in figure 4.2. If we assume, the mesh data is organised as depicted in figure 4.3 this implies, we only need to scan the visible elements of

Figure 4.15: General outline of the mesh merging procedure

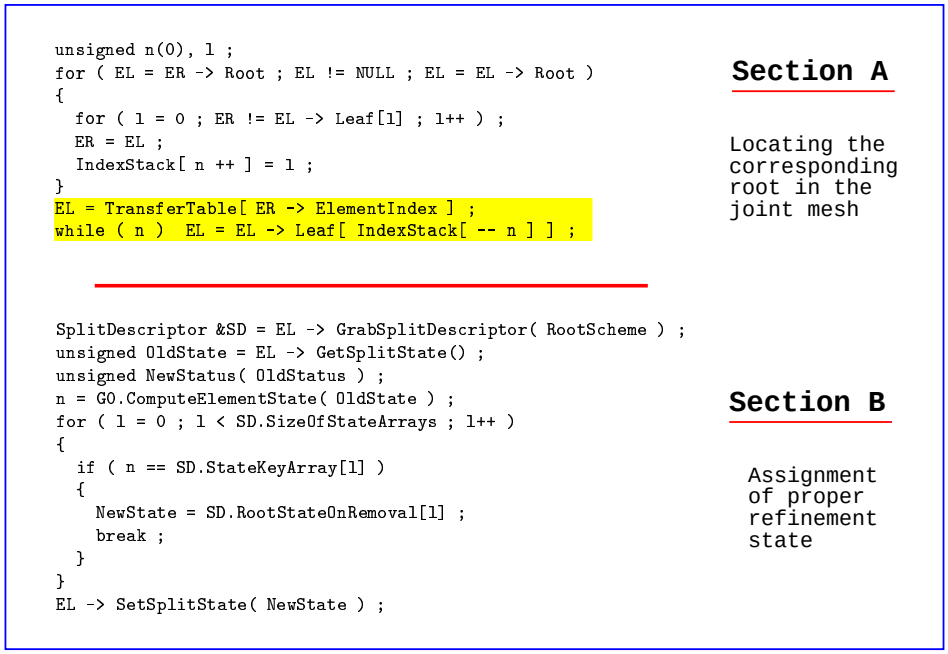
```
void FemGrid::MergeMeshes( FemGrid &G0, FemElement **TransferTable )
{
  unsigned IndexStack[ FEM_MAX_NUMBER_OF_LEVELS ] ;
  unsigned **StateStack[ FEM_MAX_NUMBER_OF_LEVELS ] ;
  unsigned k = G0.LevelData[0] -> EndOfLevel ;
  while ( k < G0.NumberOfActiveElements )
  {
    FemElement *EL, *ER = G0.ElementArray[ k ++ ] -> Root ;
    if ( ER -> TestAuxiliaryRefinement() )
    {
      unsigned RootScheme = ER -> GetSplitScheme() ;

                  . . .   < Section A >   . . .

                  . . .   < Section B >   . . .

      if ( EL -> Leaf == NULL || NewState == OldState )
        continue ;
      unsigned l, lmax, *LeafState, StackPointer(1) ;
      goto StateUpdate ;

                  . . .   < Section C >   . . .

    }
  }
}
```

the mesh G_0, which are stored consecutively at the beginning of the element array. The bottom layer cannot contain hanging nodes by construction, whence we may skip the very first elements belonging to it. Within the loop over all the relevant elements the state of the associated root is inquired into. If the presence of any special refinement pattern is detected by calling the method `FemElement::TestAuxiliaryRefinement`, the corresponding element in the joint mesh has to be found, a task delegated to section A. The refinement scheme of the root element is cached, as it is required in section B to compute the proper state of the corresponding element in the joint mesh as referenced by the handle `EL`. As the root is visited several times depending on the number of its leaves, there is a certain amount of redundancy, which could be avoided, if the root element were tagged on the successful completion of section C and elements thus marked were skipped. For simplicity, such a test has not been implemented. However, checking whether the state of the element `EL` is going to change at all serves the same purpose almost as efficiently.

In figure 4.16 both the implementation of section A and of section B are presented. Section A consists of two parts: First a path from the element referenced by the handle `ER` to the bottom layer of the mesh is established. Hereby, the handle `EL` serves as a cursor and keeps track of our current location within the hierarchy of meshes, while the handle `ER` indicates, which of the leaves we have previously visited. The leaf index is determined in the innermost loop and stored in the array termed `IndexStack`. The variable `n` is used as a stack pointer. The second part of section A is shown against a backdrop: Using the index stack that has been created in the first part of the section, the path from the element `ER` towards the bottom layer is retraced. As the starting point has been switched using the externally supplied reference table `TransferTable` we can thus reach the very element, from which we must start updating the state engines. On the completion of section A this pivot is referenced by the handle `EL`.

The implementation of the stage engine is straightforward. For each refinement scheme there is a matrix of unsigned integers, which represent the states of the newly formed leaf elements. Each row corresponds to a state, the element may have assumed, while each column corresponds to one of the leaves. The current state of the element is matched against a vector `StateKeyArray` of all those states, that lead to some state transition when the element is refined. In principle, section B imitates the refinement of the pivot element using the refinement method indicated by `RootScheme`. When the refinement pattern is removed, a state transition must be accounted for. Thereby, a new key is read from a second array, termed `RootStateOnRemoval`. If no transition

Figure 4.16: Preparing the update of the element states



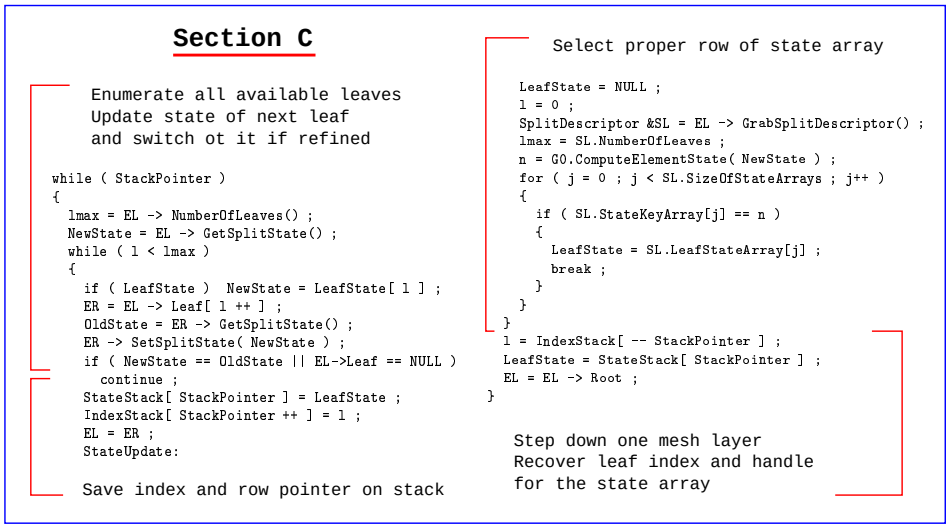
is defined, the old element state is propagated to the leaves. Should the key of the pivot remain unchanged, section C is skipped. In this way, redundant state updates are prevented.

The modified state of the pivot is stored in the variable `NewState`. It is needed to compute the key `n` employed in singling out the proper row in the state matrix, that is associated with the refinement scheme of the element `EL`. The procedure has already been detailed, when section B was discussed. A pointer to the selected row is cached in the variable `LeafState`. If no fitting row can be found, the pointer keeps its default value `NULL`. In the following the leaves of the element `EL` are enumerated. To each leaf, referenced by the handle `ER`, a new refinement state is assigned. If the old and the new state coincide the rest of the loop is skipped and the next leaf is considered. The same happens, if the element `ER` is unrefined and thence no further state updates need to be propagated to higher level elements. In any other case both the current leaf index and the pointer `LeafState` are pushed on stacks. The index and the pointer are set to their respective defaults, while the cursor `EL` is updated, such that the algorithm can be restarted with the element `EL` being replaced by its leaf `ER`. Once the number of leaves `lmax` is exhausted, the old leaf index and the old vector of refinement states are recovered from their respective stacks. The number of leaves and the state of the root are restored to the variables `lmax` respectively `OldState`. Once the handle `EL` has been replaced by `EL->Root`, the iteration is resumed with the leaf number `l`. If the stacks are empty, all elements on top of the pivot requiring an update of their refinement state have been visited. The flow of control leaves section C.

4.2.4 *On the Insertion of auxiliary Refinement Patterns*

In the previous section we have discussed a technique to generate a mesh which will eventually contain all the edges and nodes that are present in either of two given meshes. We have shown how the bulk of the emerging joint mesh can be formed and how information about auxiliary refinement patterns on the surface of one of these meshes can be propagated through the bulk of the joint mesh. The final step in the generation of the joint mesh consists in placing new special refinement patterns onto the surface of the joint mesh to ensure, that the resulting discretisation is consistent. The main difficulty with this procedure lies in the fact, that we must not assume to find a fitting refinement pattern for all of the geometrical constellation we may encounter. In the following let us therefore assume, that we can always remedy the lack of a suitable refinement

Figure 4.17: The last stage of the mesh merging procedure



scheme by splitting the problematic elements in a conventional fashion, thus simplifying the topology of the mesh. Basically all refinement strategies commonly used meet our requirement.

If we split those elements for which we cannot supply a special refinement pattern, our changes to the topology of the mesh will affect all adjoining elements on the same refinement level but one. Furthermore, additional elements may be created on lower mesh layers due to the recursive nature of the splitting algorithm outlined in figure 4.6. All of these elements have to be visited again, whenever an element has been split in a regular way. Hence, the allocation of any special refinement patterns must proceed in several passes. However, we should not place any special patterns onto the surface of the mesh against the possibility, that these patterns may be removed in a subsequent pass. To alleviate the above difficulty we can process the visible elements of the joint mesh layer by layer starting with the highest refinement level: by the above reasoning any changes to the mesh due to lacking refinement patterns will impact elements on at most the same layer only. Still, we have to account for additional elements being inserted on the *current* mesh layer, while the special refinement patterns are being assigned.

We conclude, that we have two technical problems to address: We have to assign special refinement patterns to those elements, which currently form the surface of the mesh, without actually inserting any auxiliary elements. If we fail to find a fitting refinement pattern, the topology of the mesh has to be changed by applying regular refinements. The second task consists in monitoring the mesh topology and repeating the assignment procedure in an economical fashion, until no more changes occur. The first problem has a very simple solution, since each element is equipped with a state engine and a bit field, in which its current refinement pattern is encoded. This latter device can be used to hold the allocated refinement pattern, until the assignment procedure terminates. The second problem is more involved. We employ a special flag termed `RedRefinement` to indicate, whether the current mesh layer has to be swept anew. Every time an element has been split in a regular fashion this flag is set; against any new sweep of the current mesh layer it is cleared. To avoid redundancies we maintain a list of all visible elements within the current layer. The list is initialised before the very first sweep is performed. Each element in the list is visited in turn with a view to finding its proper refinement pattern based on geometrical considerations only. If a fitting pattern is found, its identifer is stored as the current refinement scheme in the designated bit field. The element is kept in the list. If the element need not be equipped with any auxiliary refinement pattern, it is removed from the list. If such a pattern cannot be provided, the element is split in a regular way. Its adjoining elements are appended to the list, since they are the only elements on the current layer, that might have been affected by the mesh update. To prevent these elements from appearing in the list several times, an additional flag from the bit field is abused. The flag is checked before an element is appended

to the list and cleared, when it has been removed. Without such a mechanism we cannot warrant the algorithm to terminate, as the list of elements to be processed can grow longer and longer due to circular chaining.

As soon as the topology of the mesh has become sufficiently simple, such that we can provide matching refinement patterns for all elements in the current layer, we must account for their respective refinement states. In principle, the necessary steps constitute section B of the method `FemGrid::MergeMeshes` as it has been described in figure 4.16: we emulate a refinement using the very scheme we have encoded in the elements's bit field and update the refinement state in accordance with the appropriate entry in the array `RootStateOnRemoval`. All elements we need to consider are contained in the list we have maintained in the previous stage of our algorithm. Hence, we may simply traverse this list and clear it afterwards against another iteration involving the next lower mesh layer.

Once all visible elements of the mesh have been processed, the proper refinement patterns can be inserted with a view to the refinement states alone. Since we have made sure, that all these patterns will fit, the insertion need not proceed recursively and can therefore be performed by a simplified version of the algorithm outlined in figure 4.6. The simplest approach consists in creating a copy of the first part of the array `FemGrid::ElementArray`, which contains handles for all visible elements (see figure 4.3). Each element handle in this copy is de-referenced in turn; the proper refined pattern is determined from the element's refinement state and encoded into the element's bit field, as this still contains the intermediate value, we have assigned according to topological information alone. When the proper refinement pattern has been found by comparing the assigned state against all possible refinement states, the method `FemGrid::AssignedSplit` is invoked, which takes the index of the refinement pattern as its only argument. In case the refinement state reads zero, no refinement is necessary and the element can be skipped.

Figure 4.18: The complete merging algorithm

```
void FemGrid::Merge( FemGrid &G1, FemGrid &G2 )
{
  FemElement **TransferTable ;
  operator=( G1 ) ;
  CreateHashTable( TransferTable ) ;
  PrepareMerge( G2, TransferTable ) ;
  MergeMeshes( G2, TransferTable ) ;
  DeleteArray( TransferTable ) ;
  PerformTopLayerRefinement() ;
}
```

Unfortunately, the method `FemGrid::PerformTopLayerRefinement`, whose functionality we have outlined in the above paragraph, is too complex to present its implementation in this section conveniently. Hence, our implementation is rendered in the appendix B along with a more detailed discussion of some pertinent technical issues.

4.3 Multilevel techniques for constrained variational problems

In 1961 the Russian mathematician Fedorenko published a paper [66] in which he proposed an iterative solver for positive definite linear systems which later would become popular under the denomination *multigrid* respectively *multilevel* solver. It took almost twenty years and the efforts of Brandt [28, 29] to establish the fame of a method, that previously was completely overlooked. We can only speculate on the reasons for this seemingly strange neglect: Fedorenko provided a proof of convergence and found the numerical complexity of his method to depend linearly on the size of the algebraic problem. It has been alleged that his estimates have been too pessimistic to encourage the implementation of a multilevel solver. We tend to think, however, that at the time of its inception the technical prerequisites were missing that could have turned the multigrid method an instant success. The method can be easily described and implemented in a recursive fashion. It requires a fairly large amount of core memory, since the algorithm accesses the data in

roving patterns. Contemporary compilers, if available at all, did not support nested function calls. Hence, the algorithm had to be reformulated as a purely iterative process - a rather cumbersome undertaking. The mainframe technology, on which alone larger problems could be tackled, was designed with a view to out of core operations and optimised for manipulating data streams. Such a design avoided the necessity of installing larger amounts of fast and incredibly expensive core memory. Accordingly, algorithms with highly local data access patterns as for instance *Multifront solvers* [75] showed a superior performance - and were much simpler to implement.

Variational inequalities require per se iterative solution techniques. Hence, with a substantial part of the competition out of the field we should expect multilevel methods to fare much better. Nevertheless, the application of such methods to constrained problems has received comparatively little attention. Without claiming to be exhaustive, we refer to [30,78,85,91,92,106]. One possible reason for this neglect may consist in the difficult analysis of numerical algorithms for even the most generic variational inequality, the so called obstacle problem. An a priori error bound for the finite element solution of such type of problem was obtained as late as 1974 [65]. A proof of convergence for the projected Gauß-Seidel iteration, which by that time had been established as the standard solver for the associated algebraic formulation, had been found only three years before [55]. The fact is worth mentioning, as the iteration scheme had been in use [56, 81] as early as 1957. Even today, no satisfactory convergence result for multigrid schemes is known in the constrained case, that is comparable to those proofs [18, 24, 26] which have been devised for unconstrained problems. The analysis in for instance [106] merely asserts, that the proposed algorithm behaves like a conventional multigrid algorithm with an uniform contraction rate, once the set of active constraints has been fixed on the finest mesh.

4.3.1 Introductory remarks on multilevel schemes

While there exists a vast amount of literature on multilevel techniques in general, comparatively few articles have been written with a view to constrained variational problems. Moreover, all of them consider the special case of box constraints only. A generic example for the type of constrained variational setting, multilevel algorithms have been proposed for, is the obstacle problem we have considered in section 2.2: Given two functions $\psi \in H^2(\Omega)$ and $f \in L^2(\Omega)$ that are defined on a bounded domain $\Omega \subset \mathbb{R}^n$ find the minimiser of the functional

$$J(v) \;:=\; \int_\Omega \left\{ \frac{1}{2} |\nabla v|^2 \;-\; f\, v \right\} dx \qquad ; \qquad v \;\in\; H_0^1(\Omega) \tag{4.4}$$

under the additional assumption: $v \geq \psi$ almost everywhere on Ω. A conforming finite element scheme which employs linear shape functions to approximate the minimiser was analysed in 1974 by Falk [65]. Today this method can still be regarded as the standard discretisation for the primal formulation. The use of quadratic shape functions in a conforming finite element framework was discussed in 1977 by Brezzi, Hager and Raviart [36]. One year later these authors proposed a mixed finite element method for the obstacle problem [37], that relied on the following dual formulation: Minimise the functional

$$J^*(\sigma) \;:=\; \int_\Omega \left\{ \frac{1}{2} \sigma^2 \;+\; \psi\, \nabla\cdot \sigma \right\} dx \qquad ; \qquad \sigma \;\in\; H_{\mathrm{div}}(\Omega) \tag{4.5}$$

under the condition: $\nabla \cdot \sigma + f \leq 0$ almost everywhere on Ω. We have seen in section 3.3.2 that the solution of the dual mixed formulation (4.5) can be very helpful in the computation of sharp hypercycle estimates for the obstacle problem. Hence, the development of a fast multilevel solvers for the problem (4.5) is definitely not a purely academic pastime.

It seems, only one paper [111] has been published which describe an actual implementation of a dual mixed discretisation scheme. The multilevel methods that have been developed to solve the primal formulation (4.4) either use suitably adapted algorithmic components of a conventional linear multigrid solver [30,106] or carry out combinatorial procedures employing linear multilevel iterations to update the set of active constraints (see for example [78,85]). The algorithm proposed in [111] falls into the latter category. However, its multigrid component is merely a device to evaluate a Schur complement resulting from a transformation [9] of the dual problem back into

a primal setting: the active set is actually updated by a simple conjugate gradient iteration. As to the efficiency of this approach no conclusive numerical experiments are reported.

Replacing the dual mixed discretisation of an elliptic differential equation by a nonconforming discretisation of its primal formulation is an artifice to be found also elsewhere (for example [31]). In the design of multilevel iterations for unconstrained problems such a substitution is appropriate, as it permits the use of established multigrid technologies. In a constrained setting, however, returning to a primal formulation only leads to a convoluted algorithm without breaking the saddle point structure of the algebraic problem. The introduction of penalty terms is fraught with difficulties even in the unconstrained case [43] and fails likewise in lifting the saddle point character of the problem. Hence, it seems reasonable to treat the dual mixed discretisation in the most immediate manner possible. While a direct approach is easily pursued with the help of Vanka-type smoothers [134] in the case of the Stokes problem, the construction of analogous smoothing iterations for the dual formulation (4.5) is inherently difficult. Elimination of the dual variables and solving the resulting complementary problem for the Lagrange multipliers of the admissibility constraints is a possible resort: but conventional projected point relaxation methods no longer qualify as smoothers, as the Schur complement matrix is dense.

Below, we will consider two technologies, which address the aforementioned difficulties. We will introduce a defect correction scheme for the Schur complement problem, that is derived from the projected SOR iteration and provides the opportunity to apply a preconditioner to the Schur complement. The new scheme may either be used as an iterative solver or as a smoother in a multilevel context. In section 4.3.4 we supply a proof of convergence to cover the first scenario. If the iteration is employed as a smoother, its performance depends on the choice of the preconditioner and on the speed, the defect can be computed with. After the vector valued shape functions have been scaled properly, a simple CG iteration is sufficient to evaluate the Schur complement with optimal complexity. To obtain an preconditioner we may lump the mass matrix and calculate the resulting matrix product explicitly. Alternatively, we may rely on a hybrid dual mixed discretisation as a means to construct the preconditioner.

The second numerical technology relies on hybridisation to define a nonlinear equation for the Lagrange multipliers that are associated with the continuity constraints on the normal fluxes. In this way, the global linear complementary condition is replaced by very small, independent complementary problems, each of them defined on one of the elements compounding the grid. Thus the computation of the defect becomes as inexpensive as a matrix vector multiplication. A defect correction scheme is devised that is accelerated by a multilevel method operating in the so called full approximation storage (FAS) mode. Due to the sparsity patterns of the operators involved any established point relaxation method can be employed to improve the smoothing.

4.3.2 Statement of the dual formulation

In the following let us use the notation we have introduced in the sections 2.1.1 and 3.2.2. Basically, we shall consider the obstacle problem in the description we have given in section 2.2.2. For the sake of simplicity we will make a number of additional assumption, however: We suppose, that the sets $\Omega \subset \mathbb{R}^n$ and $\Omega_0 \subseteq \Omega$ are both polygonal and that Ω is bounded and convex. The obstacle function $\psi \in H^1(\Omega)$ we assume to be piecewise linear, that is: $\psi \in P^1(\mathcal{M}_h)$. Hereby $\mathcal{M}_h$ may denote a simplicial decomposition of the domain Ω, such that complementary subsets of its elements form meshes for the domains Ω_0 and $\Omega \setminus \Omega_0$ as well. The primal formulation reads:

Problem P: *The equilibration position of an elastic "membrane", represented by the domain Ω, is described by a function $u^* \in H^1(\Omega)$, which denotes its distortion in normal direction under the action of some load $f \in W^{1,1}(\Omega)$. It is assumed, the membrane is obstructed by an "obstacle" $\psi \in P^1(\mathcal{M}_h)$ and has been clamped along $\partial\Omega$ to adopt the values of some function $u_0 \in W^{2,\infty}(\Omega)$ with the property: $u_0 > \psi$ on $\partial\Omega$. The material law is described by the components $A_{ij} \in L^\infty(\Omega)$ of a symmetric tensor, that satisfies the ellipticity condition*

$$A_0 > 0 \quad : \quad \sum_{i=1}^{n} \sum_{j=1}^{n} A_{ij}(x)\, \xi_i\, \xi_j \;\geq\; A_0\, \|\xi\|^2 \quad ; \quad \xi \in \mathbb{R}^n\,,\; x \in \Omega\;. \tag{4.6}$$

The cone $V_\psi \subset H^1(\Omega)$ of possible equilibria is defined by:

$$V_\psi \quad := \quad \left\{ u \in H^1(\Omega) \,\middle|\, v \in H_0^1(\Omega) \;:\; u \,=\, v + u_0 \,\geq\, \psi \;\; \text{on } \Omega_0 \right\} \;.$$

If the elastic energy $J : H^1(\Omega) \longrightarrow \mathbb{R}$ of the membrane is modelled by the functional

$$J(v) \quad := \quad \frac{1}{2} \int\limits_\Omega \left\{ (\nabla v)^T A \, \nabla v \,-\, 2\, f\, v \right\} \, d\,x \qquad ; \qquad v \,\in\, H^1(\Omega)$$

the equilibrium position $u^* \in V_\psi$ must obey Hamilton's principle in the form:

$$\int\limits_\Omega (\nabla u^*)^T A \, (\nabla v - \nabla u^*) \; d\,\xi \quad \geq \quad \int\limits_\Omega f \, (\, v \,-\, u^*) \; d\,\xi \qquad ; \qquad v \in V_\psi \;\; .$$

□

Due to condition (4.6) the inverse of the matrix $A(x)$ exists for every $x \in \Omega$. Let $\mathfrak{M}_\psi$ denote the set of all positive measures, whose support is confined to the set Ω_0. (The notation has been introduced in section 2.2.3). Furthermore, define $\mathfrak{L} : H^1(\Omega) \times L^2(\Omega, \mathbb{R}^n) \times \mathfrak{M}_\psi \longrightarrow \mathbb{R}$ by:

$$\mathfrak{L}(v, \tau, \mu^*) \quad := \quad \int\limits_\Omega \left\{ \, \tau \, \nabla v \,-\, \frac{1}{2} \, \tau^T A^{-1} \tau \,-\, f \, v \, \right\} \; d\,\xi \quad - \quad \langle \, v - \psi, \, \mu^* \, \rangle_{\Omega_0} \;\; .$$

In section 2.2.5 it has been demonstrated, that the elastic energy $J(v)$ of the membrane can be recovered by computing the supremum of the Lagrangian $\mathfrak{L}$ with respect to its second and third argument. The complementary energy functional $J^* : H_{\text{div}}(\Omega) \longrightarrow \mathbb{R}$ may be obtained by finding the infimum of the Lagrangian $\mathfrak{L}$ with respect to its first argument. We define:

$$J^*(\tau, \mu^*) \quad := \quad \begin{cases} \displaystyle\int\limits_\Omega \left\{ \, \tau \, \nabla u_0 \,-\, \frac{1}{2} \, \tau^T A^{-1} \tau \, \right\} \, d\,\xi \;+\; \langle \, \psi - u_0, \, \mu^* \, \rangle_{\Omega_0} & ; \quad \tau \in Q^*_{\mu^*} \\ -\,\infty & ; \quad \text{else} \end{cases}$$

with the set of admissible vector fields $Q^*_{\mu^*} \subset L^2(\Omega, \mathbb{R}^n)$ being specified by (2.17). Due to the regularity inherent in the primal formulation we may restrict our attention to vector fields from the space $H_{\text{div}}(\Omega)$. Carrying out a partial integration we attain for any $\tau \in Q^*_{\mu^*} \cap H_{\text{div}}(\Omega)$:

$$J^*(\tau, \mu^*) \;\; = \;\; \oint\limits_{\partial\Omega} u_0 \, \tau_n \; d\,s \;\; - \;\; \int\limits_\Omega \left\{ \, \psi \, \nabla \cdot \tau \,+\, \frac{1}{2} \, \tau^T A^{-1} \tau \, \right\} \; d\,\xi \;\; - \;\; \int\limits_\Omega f \, (\psi - u_0) \; d\,\xi \quad .$$

Hereby, the subscript n denotes the normal component of the vector field τ with respect to the hypersurface $\partial\Omega$. Since $\nabla \cdot \tau$ is a square integrable function by assumption, this quantity is well defined as a distribution acting on certain traces (see e. g. theorem 2.5 in [76]). Dropping any constant expressions we may now identify the complementary energy as:

$$J^*(\tau) \;\; = \;\; \int\limits_\Omega \left\{ \, \psi \, \nabla \cdot \tau \,+\, \frac{1}{2} \, \tau^T A^{-1} \tau \, \right\} \; d\,\xi \;\; - \;\; \oint\limits_{\partial\Omega} u_0 \, \tau_n \; d\,s \quad ; \quad \tau \in H_{\text{div}}(\Omega) \;\; .$$

The dual formulation of problem **P** demands the minimisation of $J^*(\tau)$ under the assumption, that the admissibility constraint $-f - \nabla \cdot \tau \in \mathfrak{M}_\psi$ be met. The solution σ^* of the dual problem is related to the equilibrium position u^* of the membrane by the duality mapping:

$$\sigma^* \quad = \quad A \, \nabla u^* \qquad .$$

Problem D: *The regularity requirements stated in the formulation of problem **P** be met and the set of admissible stresses be defined by:*

$$S \quad := \quad \left\{ \, \tau \in H_{\text{div}}(\Omega) \,\middle|\, -\nabla \cdot \tau \,-\, f \,\in\, \mathfrak{M}_\psi \, \right\} \;.$$

The equilibrium distribution σ^* of the stress inside the membrane is the solution of the following constrained variational problem: Find $\sigma^* \in S$, such that:

$$\int\limits_\Omega (A^{-1} \sigma^*)^T (\, \tau - \sigma^* \,) \; d\,\xi \;+\; \int\limits_\Omega \psi \, (\, \nabla \cdot \tau - \nabla \cdot \sigma^* \,) \; d\,\xi \;\; \geq \;\; \oint\limits_{\partial\Omega} u_0 \, (\, \tau_n - \sigma^*_n \,) \; d\,s \quad ; \quad \tau \in S \;.$$

□

4.3.3 Discretisation of the dual formulation

Once the mesh $\mathcal{M}_h$ has been supplied, various trial spaces can be constructed, in which we will look for the numerical solution of problem **D**. These spaces have already been introduced in section 3.2.2. However, we have not yet dealt with the more intricate aspects of assembling the finite element matrices. To specify the degrees of freedom associated with our trial spaces test functions must be considered that live on the interfaces between adjacent simplices. We define:

$$\begin{aligned} \mathcal{I}_h &:= \left\{ E \subset \Omega \;\middle|\; T, T' \in \mathcal{M}_h \,:\, E = T \cap T' \,\wedge\, T \neq T' \right\} \quad , \\ \mathcal{B}_h &:= \left\{ E \subset \partial\Omega \;\middle|\; T \in \mathcal{M}_h \,:\, E = T \cap \partial\Omega \right\} \quad . \end{aligned}$$

The symbol $\mathcal{I}_h$ denotes the set of all interior interfaces between adjacent simplices, while $\mathcal{B}_h$ designates the set of all those faces, which constitute the boundary $\partial\Omega$ of the computational domain. The spaces $P^k(\mathcal{M}_h)$ and $P^k(\mathcal{I}_h \cup \mathcal{B}_h)$ are of Lagrange type: their degrees of freedom consist of function evaluations at certain collocation points $\hat{x}^{(k)}_{T,j} \in \Omega$ (see [48] §2.2 for details). By $u^{(k)}_{T,j} \in P^k(\mathcal{M}_h)$ we shall denote the corresponding shape functions. For any element $T \in \mathcal{M}_h$ and any of its faces $E \subset \partial T$ we can now define a set of linear functionals:

$$\rho^{(k)}_{T,E,j}(\tau) \quad := \quad \int\limits_E u^{(k)}_{T,j}\, \tau_n \; d\,s \quad ; \quad \tau \in H_{\mathrm{div}}(\Omega) \quad . \tag{4.7}$$

Let $e_l \in \mathbb{R}^n$ represent the l-th basis vector of a Cartesian coordinate system. Against the case $k > 0$ we introduce a second set of functionals:

$$\rho^{(k)}_{T,j,l}(\tau) \quad := \quad |T|^{-1/d} \int\limits_T u^{(k-1)}_{T,j}\, e_l \cdot \tau \; d\,\xi \quad ; \quad \tau \in H_{\mathrm{div}}(\Omega) \quad . \tag{4.8}$$

Both sets together constitute the degrees of freedom associated with the ansatz $R^k(\mathcal{M}_h)$. The functionals (4.7) are shared by the spaces $B^k(\mathcal{M}_h)$. In the case $k > 1$ the other degrees of freedom differ. Let $\{\sigma^{(k)}_{T,i}\}_{T,i}$ denote a basis of the function space $B^k_0(\mathcal{M}_h) := \{\tau \in B^k(\mathcal{M}_h) \,|\, \nabla\cdot\tau = 0\}$ such that $\|\sigma^{(k)}_{T,i}\|_\infty = 1$ holds for each basis function. Instead of (4.8) the missing degrees of freedom for the trial space $B^k(\mathcal{M}_h)$ now read:

$$\begin{aligned} \rho^{(k)}_{T,j}(\tau) \quad &:= \quad \int\limits_T \nabla u^{(k)}_{T,j} \cdot \tau \; d\,\xi \quad ; \quad \tau \in H_{\mathrm{div}}(\Omega) \quad , \\ \hat{\rho}^{(k)}_{T,i}(\tau) \quad &:= \quad |T|^{-1/d} \int\limits_T \sigma^{(k)}_{T,i} \cdot \tau \; d\,\xi \quad ; \quad \tau \in H_{\mathrm{div}}(\Omega) \quad . \end{aligned}$$

Neither $B^k(\mathcal{M}_h)$ nor $R^k(\mathcal{M}_h)$ are subspaces of $H_{\mathrm{div}}(\Omega)$: Yet a vector field $\sigma^{(k)}$ from either of these spaces is contained in $H_{\mathrm{div}}(\Omega)$, if and only if the normal fluxes of $\sigma^{(k)}$ are continuous across the element interfaces inside the domain Ω. Hence, there are two possibilities to obtain a conforming numerical approximation to the solution of problem **D**. Since our choice of the ansatz does not materially alter the technical procedure, we will discuss these possibilities by the example of the space $R^k(\mathcal{M}_h)$ only. Let $\{\sigma^{(k)}_{T,E,j}, \sigma^{(k)}_{T,j,l}, \ldots\} \subset R^k(\mathcal{M}_h)$ denote the dual basis of (4.7) and (4.8). Our first option is to construct a basis $\{\sigma^{(k)}_{E,j}, \sigma^{(k)}_{T,j,l}, \ldots\}$ for the conforming trial space

$$R^k_c(\mathcal{M}_h) \quad := \quad R^k(\mathcal{M}_h) \;\cap\; H_{\mathrm{div}}(\Omega) \quad .$$

To this end we may specify a weight function $\omega : \mathcal{M}_h \times (\mathcal{I}_h \cup \mathcal{B}_h) \longrightarrow \{-1, 0, 1\}$, such that

$$\sigma^{(k)}_{E,j} \quad := \quad \sum_{T \in \mathcal{M}_h} \omega(T,E)\; \sigma^{(k)}_{T,E,j} \qquad \in \qquad H_{\mathrm{div}}(\Omega) \tag{4.9}$$

holds. The case $\omega(T,E) = 0$ is implied by $E \not\subset \partial T$. Let $\mathcal{M}^0_h \subset \mathcal{M}_h$ denote the very part of the mesh, which corresponds to the contact area Ω_0. To account for the admissibility constraint $\sigma \in S$, a cone of test functions is defined by:

$$\Lambda_k \quad := \quad P^k(\mathcal{M}_h) \;\cap\; L^2(\Omega_0, \mathbb{R}^+_0) \quad . \tag{4.10}$$

Furthermore, a Lagrangian $L_c^k : R^k(\mathcal{M}_h) \times \Lambda_k \longrightarrow \mathbb{R}$ is introduced:

$$L_c^k(\sigma, v) \quad := \quad \sum_{T \in \mathcal{M}_h} \int_T \Big\{ \frac{1}{2}\, \sigma^T A^{-1} \sigma \; + \; \psi\, \nabla\cdot \sigma \; + \; v\, (\, \nabla\cdot \sigma + f\,) \Big\}\, d\,\xi \;\; - \;\; \sum_{E \in \mathcal{B}} \int_E u_0\, \sigma_n \, d\, s \; .$$

The solution of problem **D** can be identified with the unique saddle point $\{\sigma_k, v_k\} \in R_c^k(\mathcal{M}_h) \times \Lambda_k$ of the above Lagrangian. Our second option consists in using additional Lagrange multipliers to enforce the continuity of the normal fluxes of $\sigma_k \in R^k(\mathcal{M}_h)$ across the element interfaces. The augmented Lagrangian $L^k : R^k(\mathcal{M}_h) \times \Lambda_k \times P^k(\mathcal{I}_h) \longrightarrow \mathbb{R}$ reads:

$$L^k(\sigma, v, w) \quad := \quad L_c^k(\sigma, v) \;\; - \;\; \sum_{E \in \mathcal{I}_h} \sum_{T \in \mathcal{M}_h} \; |\, \omega(T, E)\, | \int_E w\, \sigma_n \;\, d\, s \quad . \tag{4.11}$$

If we denote the set of linear factors for vector field σ_k by $s \in \mathbb{R}^{m_1}$, that for the displacement v by $\mu \in \mathbb{R}^{m_2}$ and that for the Lagrange multiplier $w \in P^k(\mathcal{I}_h)$ by $\lambda \in \mathbb{R}^{m_3}$, we may write the first order optimality condition for the saddle point $\{s, \mu, \lambda\}$ of the Lagrangian (4.11) as:

$$\left| \begin{array}{rcrcl} M\, s & + & D^T\, \mu & - & F^T\, \lambda \;\; = \;\; t \\ D\, s & + & & & I_0\, r \;\; = \;\; d \\ F\, s & & & & = \;\; 0 \end{array} \right| \quad . \tag{4.12}$$

If we employ the conforming discretisation scheme, the field σ_k must be represented by a different set $s_c \in \mathbb{R}^{m_4}$ of linear factors. The first order optimality condition for the saddle point $\{s_c, \mu\}$ of the Lagrangian $L_c^k(\sigma, v)$ reads:

$$\left| \begin{array}{rcrcl} M_c\, s_c & + & D_c^T\, \mu & = & t_c \\ D_c\, s_c & + & I_0\, r & = & d \end{array} \right| \quad . \tag{4.13}$$

Hereby, the symbol I_0 denotes the finite element matrix associated with the injection operator $I : P^k(\mathcal{M}_h^0) \longrightarrow P^k(\mathcal{M}_h)$ which extends all test functions outside the domain Ω_0 by 0. We note, that in the case $k \in \{0, 1\}$ the requirement $v \in \Lambda_k$ gives rise to a very simple linear complementary condition for the vectors $r \in \mathbb{R}^{m_5}$ and μ:

$$I_0^T \mu\, , \; r \quad \geq \quad 0 \qquad \wedge \qquad \mu^T (I_0\, r) \quad \stackrel{!}{=} \quad 0 \quad .$$

Remark 4.4 In the case $k \geq 2$ the requirement $v \in \Lambda_k$ no longer translates into the simple algebraic condition $I_0^T \mu \geq 0$. Hence, additional inequality constraints appear to account for (4.10). Moreover, testing with $v \in \Lambda_k$ no longer warrants $\sigma_k \in S$, even if some load $f \in P^k(\mathcal{M}_h)$ is chosen: a consistency error is thus introduced.

4.3.4 Description of the multilevel algorithm

In the case $k \in \{0, 1\}$ the condition $I_0^T \mu \geq 0$ is equivalent to enforcing $\mu_i \geq 0$ for a certain subset $0 < i_1 < \ldots i_m \leq m_2$ of indices. Let us denote this subset by $\mathfrak{I}_0$. To simplify the treatment of the admissibility condition $\sigma_k \in S$, we may now introduce box constraints $\mu \geq \psi$ with a suitable obstacle $\psi \in \mathbb{R}^{m_2}$. Picking a sufficiently small value $\mu_0 \ll 0$ we may define:

$$\psi_i \quad := \quad \left\{ \begin{array}{lcl} 0 & ; & i \in \mathfrak{I}_0 \\ \mu_0 & ; & \text{else} \end{array} \right. \quad .$$

If we now eliminate those linear factors from either (4.12) or (4.13), which are associated with the vector field σ_k, a linear complementary problem for the Lagrange multiplier $v \in \Lambda_k$ of the admissibility constraint $-\nabla\cdot \sigma_k - f \in \mathfrak{M}_\psi$ eventually emerges:

$$\begin{array}{rcrcl} W\, \mu & := & C\, \mu \;\; + \;\; B\, A^{-1}\, B^T\, \mu & \geq & g \\ \wedge \quad \mu & \geq & \psi \quad \wedge \quad (\, \mu - \psi\,)^T (\, W\, \mu \; - \; g\,) & = & 0 \quad . \end{array} \tag{4.14}$$

If a hybrid discretisation scheme is used, the matrices A, B and C correspond to:

$$A \;:=\; -FM^{-1}F^T \quad , \qquad B \;:=\; DM^{-1}F^T \quad , \qquad C \;:=\; DM^{-1}D^T \quad .$$

If a conforming discretisation is employed instead, these matrices are defined by:

$$A \;:=\; M_c \quad , \qquad B \;:=\; D_c \quad , \qquad C \;:=\; 0 \quad .$$

As the trial spaces $R^k(\mathcal{M}_h) \times P^k(\mathcal{M}_h)$ and $B^k(\mathcal{M}_h) \times P^{k-1}(\mathcal{M}_h)$ meet the *LBB-condition* (see [34, 54]), the matrix W is positive definite both in the hybrid and in the conforming case. Hence, the problem (4.14) features an unique solution (see e. g. [129]). By $\texttt{pSOR}(W, \psi, g, \mu, \omega)$ we shall designate the result of one sweep of the projected SOR-scheme as it is described in [55]. Hereby, W denotes the matrix under consideration, ψ the obstacle, g the right hand side, μ the initial guess and $\omega \in (0, 2)$ the relaxation parameter. By

$$\mu^{(j+1)} \;:=\; \texttt{pSOR(}\, W,\, \psi,\, g,\, \mu^{(j)},\, \omega\, \texttt{)} \qquad ; \qquad j \in \mathbb{N} \tag{4.15}$$

a smoothing iteration may be defined. However, the matrix W is dense in both the hybrid and the conforming case, even though A, B and C themselves are very sparse. Therefore, the iteration (4.15) is too expensive. However, if the constraint $\mu^{(n)} \geq \psi$ did never become active, the smoother (4.15) would be equivalent to the following iteration:

$$\mu^{(j+1)} \;:=\; \mu^{(j)} \;+\; \texttt{pSOR(}\, W,\, \psi \,-\, \mu^{(j)},\, g - W\mu^{(j)},\, 0,\, \omega\, \texttt{)} \qquad ; \qquad j \in \mathbb{N} \quad .$$

This latter rule can be understood as defining a defect correction scheme with the matrix W acting as a preconditioner. Consequently, it seems reasonable to replace W by a sparse approximation $\hat{W}$. To ensure that the resulting fixed point iteration is still convergent, damping parameters $\Theta > 0$ and $\hat{\Theta} \in (0, 1]$ must be introduced:

$$\mu^{(j+1)} \;:=\; \mu^{(j)} \;+\; \hat{\Theta}\, \texttt{pSOR(}\, \hat{W}, \psi - \mu^{(j)}, \Theta\,(g - W\mu^{(j)}), 0, \omega\, \texttt{)} \qquad ; \qquad j \in \mathbb{N} \quad . \tag{4.16}$$

As we will show in subsection 4.3.5, the above iteration yields a solver for the complementary problem (4.14), if the preconditioner $\hat{W}$ is positive definite and the product $\Theta\hat{\Theta}$ sufficiently small. The matrix A is essentially a mass matrix both in the conforming and in the hybrid case. Hence, its condition number $\kappa(A)$ is uniformly bounded in h, the grid parameter. In consequence, the residual $g - W\mu^{(j)}$ can be evaluated with the help of a simple CG iteration and the numerical complexity of this procedure is still optimal.

In default of a canonical approach to approximate A by a diagonal matrix $\hat{A}$ the preconditioner $\hat{W}$ may be supplied either by lumping A or by dropping all its entries off the diagonal. In the hybrid case the resulting matrix need not necessarily be positive definite, however. Hence, it is convenient to introduce a third damping parameter $\tilde{\Theta} > 0$ and define the preconditioner by:

$$\hat{W} \;:=\; C \;+\; \tilde{\Theta}\; B\hat{A}^{-1}B^T \quad .$$

This matrix can be computed and stored with optimal complexity $\mathcal{O}(m_2)$, as the number of entries per row can be bounded by a power of k, the order of the trial space, times $n + 1$.

We will assume, a regular family $\{\mathcal{M}_{h_l}\}_{l>0}$ of meshes has been generated in such a way, that the trial spaces for the Lagrange multiplier μ are nested: $P^k(\mathcal{M}_{h_1}) \subseteq P^k(\mathcal{M}_{h_2}) \subseteq \ldots P^k(\mathcal{M}_h)$. For each refinement level l an imbedding operator $\iota_l : P^k(\mathcal{M}_{h_l}) \longrightarrow P^k(\mathcal{M}_{h_{l+1}})$ is introduced. The matrix I_l associated with the imbedding ι_l consists of the following entries:

$$I^{(l)}_{Ti,Uj} \;:=\; u^{(k)}_{U,j}(\hat{x}^{(k)}_{T,i}) \qquad ; \qquad U \in \mathcal{M}_{h_l}\,, \;\; T \in \mathcal{M}_{h_{l+1}} \quad . \tag{4.17}$$

To each map ι_l corresponds a restriction operator $\varrho_l : P^k(\mathcal{M}_{h_{l+1}}) \longrightarrow P^k(\mathcal{M}_{h_l})$ we must define in such a way, that the coarse grid correction step of our multilevel algorithm warrants an admissible approximation. We find we have to insist on:

$$(\iota_l \circ \varrho_l)^2\,(v) \;=\; (\iota_l \circ \varrho_l)\,(v) \;\geq\; v \qquad ; \qquad v \in P^k(\mathcal{M}_{h_{l+1}}) \quad .$$

No linear operator ϱ_l can be discovered which meets this requirement. However, in the special case $I_l \geq 0$ a suitable map can be found easily: let $\{\mu_{Ti}^{(l+1)}\}_{Ti}$ denote the set of linear factors of some function $v^{(l+1)} \in P^k(\mathcal{M}_{h_{l+1}})$ - the factors of the image $\varrho_l(v^{(l+1)})$ may be computed by:

$$\mu_{Uj}^{(l)} \;:=\; \max_{T,i} \left\{ \mu_{Ti}^{(l+1)} \;\middle|\; I_{Ti,Uj}^{(l)} \;>\; 0 \right\} \quad . \tag{4.18}$$

Depending on the desired level of accuracy we fix a mesh $\mathcal{M}_{h_L}$, on which the solution of problem **D** is to be approximated. On each of the meshes $\mathcal{M}_{h_1}, \ldots \mathcal{M}_{h_L}$ several matrices must be assembled that define the complementary problem: these will be denoted by $\{A_l, B_l, C_l\}_{l=1}^{L}$. The amount of smoothing to be applied to the data may differ depending on the level l. A set of counters $\{N_l\}_{l=1}^{L}$ specifies how many times in succession relaxations be carried out. On the mesh $\mathcal{M}_{h_L}$ with the highest refinement level the algebraic accuracy is monitored, the fixed point of the iteration (4.16) has been approximated with. Since this quantity cannot be measured by any norm of the residual, a viable alternative [30] is to control the distance between subsequent iterates by means of a parameter ϵ. If the test for convergence fails, another multigrid cycle must be initiated. An overview of the entire algorithm is presented in figure 4.19.

Figure 4.19: Outline of the Multigrid Algorithm (pMG)

Supplied: $\{A_l, B_l, C_l, S_l, \Theta_l, \hat{\Theta}_l, \omega_l, M_l, \}_{l=0}^{L}$, $\{I_l\}_{l=0}^{L-1}$, $\epsilon > 0$

Definition: pMG(μ, f, ψ, l) {

1) Set: $\mu_{\mathtt{old}} := \mu$.
2) Loop $m \in \{1 \ldots M_l\}$: {
2a) Apply CG to solve: $A_l\, v \;=\; B_l^T\, \mu$.
2b) Set: $r \;:=\; \Theta_l\,(\, f \;-\; C_l\, \mu \;-\; B_l\, v\,)$.
2c) Loop i : {
$\hat{\eta}_i \;:=\; \dfrac{1}{S_{ii}} \left(r_i \;-\; \sum_{j<i} S_{ij}\, \eta_j \right)$.
$\eta_i \;:=\; \max\left\{ \psi_i - \mu_i,\; \omega_l\, \hat{\eta}_i \right\}$.
}
2d) Set: $\mu \;:=\; \mu \;+\; \hat{\Theta}_l\, \eta$.
}
3) If $l = 0$ return.
4) Apply CG to solve: $A_l\, v \;=\; B_l^T\, \mu$.
5) Set: $\tilde{f} \;:=\; I_{l-1}^T\,(\, f \;-\; C_l\, \mu \;-\; B_l\, v\,)$.
6) Compute $\tilde{\psi}$ from $\psi - \mu$ e.g. by (4.18).
7) Set: $\tilde{\mu} \;:=\; 0$.
8) Call: pMG($\tilde{\mu}$, $\tilde{f}$, $\tilde{\psi}$, $l-1$).
9) Set: $\mu \;:=\; \mu \;+\; I_{l-1}\, \tilde{\mu}$.
10) Repeat step 2.
11) If $l < L$ or $\| \mu_{\mathtt{old}} - \mu \|_\infty < \epsilon$ return.
12) Goto step 1.
}

4.3.5 A proof of convergence

As the title indicates, we are going to present a proof, that the iteration defined by (4.16) is convergent to the solution of the linear complementary problem (4.14) for any positive definite

preconditioner and any relaxation parameter $\omega \in (0,2)$, if only the product $\Theta\hat{\Theta}$ of the damping parameters is small enough. Since we will need not invoke any results from previous sections, let us adopt the following, more conventional notation: $A \in \mathbb{R}^{m\times m}$ and $S \in \mathbb{R}^{m\times m}$ may denote two positive definite matrices. Furthermore, let $\psi \in \mathbb{R}^m$ and $f \in \mathbb{R}^m$ designate two arbitrary vectors. We recall, that the solution of the linear complementary problem

$$A\,x \;\geq\; f \qquad \wedge \qquad x \;\geq\; \psi \qquad \wedge \qquad (\,x-\psi\,)^T(\,A\,x\,-\,f\,) \;=\; 0 \quad . \tag{4.19}$$

is uniquely defined. Henceforth, we will refer to this solution as $x^* \in \mathbb{R}^m$. The defect correction scheme specified by (4.16) is detailed in figure 4.19. Let us write the iteration as

$$x^{(j+1)} \;:=\; x^{(j)} \;+\; \hat{\Theta}\; T_\Theta(x^{(j)}) \qquad ; \qquad j \in \mathbb{N} \quad . \tag{4.20}$$

Hereby, the symbol T_Θ represents a nonlinear map $T_\Theta : \mathbb{R}^m \longmapsto \mathbb{R}^m$, which may be described by:

$$\begin{aligned}
&\texttt{1) Set: } r \;:=\; \Theta\,(\,f\,-\,A\,x\,)\,.\\
&\texttt{2) Loop } i\texttt{: \{}\\
&\qquad\qquad \hat{y}_i \;:=\; r_i \,-\, \sum_{j<i} S_{ij}\,y_j \;.\\
&\qquad\qquad y_i \;:=\; \max\Big\{\,\psi_i - x_i,\; \frac{\omega}{S_{ii}}\;\hat{y}_i\,\Big\}\;.\\
&\qquad \texttt{\}}\\
&\texttt{3) Set: } \;\; T_\Theta(x) \;:=\; y\,.
\end{aligned} \tag{4.21}$$

Proposition 4.1 *For any damping parameter $\Theta > 0$ and any relaxation parameter $\omega > 0$ we find $T_\Theta(x) = 0$, if and only if $x = x^*$ holds.*

Proof Let us assume $T_\Theta(x) = 0$ holds for some vector $x \in \mathbb{R}^m$. We are going to demonstrate, that this very vector is the solution of problem (4.19). With a view to (4.21) we conclude:

$$0 \;\;=\;\; \max\Big\{\,\psi_i - x_i,\; \frac{\omega}{S_{ii}}\;r_i\,\Big\} \qquad ; \qquad i \in \{1,\ldots m\} \quad .$$

From the above set of equations we immediately infer, that $x \geq \psi$ and $r \leq 0$ hold. We now have to show, that the complementary condition $(x-\psi)^T r = 0$ is also met. Let us suppose, there is an index j and some number $\varepsilon > 0$, such that $r_j < -\varepsilon$ and $x_j - \psi_j > \varepsilon$ both hold. We find:

$$0 \;\;=\;\; \max\Big\{\,\psi_j - x_j,\; \frac{\omega}{S_{jj}}\;r_j\,\Big\} \;\;\leq\;\; \max\Big\{\,-\varepsilon,\; -\frac{\omega}{S_{jj}}\;\varepsilon\,\Big\} \;\;=\;\; -\,\varepsilon\;\min\Big\{\,1\,,\;\frac{\omega}{S_{jj}}\,\Big\} \quad .$$

The above contradiction proves $x = x^*$. We are now going to show $T_\Theta(x^*) = 0$. To this end we recall, that x^* satisfies the complementary condition $(x^*-\psi)^T(Ax^*-f) = 0$. Hence, we note $x^*_i = \psi_i$ whenever $r_i < 0$ holds. We will carry out a proof by induction: $r_1 = 0$ implies $\hat{y}_1 = 0$ and thence $y_1 = 0$. If we assume $r_1 < 0$, we conclude $\hat{y}_1 < 0$ and $\psi_1 - x^*_1 = 0$. Therefore, we also find $y_1 = 0$. Let us suppose, we have discovered $y_1 = \ldots y_{i-1} = 0$ for some index $i > 1$. As $\hat{y}_i = r_i$ holds by assumption, there are again two possibilities: both lead to $y_i = 0$ for the very same reasons we have stated in the case $i = 1$. □

As a corollary of the above proposition we may claim, that the iteration (4.20) can have but one fixed point, namely x^*. We proceed by demonstrating that the sequence $\{x^{(j)}\}_{j\in\mathbb{N}}$ generated from some vector $x^{(0)} \geq \psi$ by (4.20) minimises a quadratic form $G : \mathbb{R}^m \longmapsto \mathbb{R}$. As the level set

$$\mathcal{G} \;:=\; \Big\{\; x \in \mathbb{R}^m \;\Big|\; G(x) \;\leq\; G(x^{(0)}) \;\Big\}$$

is compact, we can thus conclude, there must be at least one cluster point $\hat{x} \in \mathcal{G}$, such that:

$$\liminf_{j\to\infty}\; G(x^{(j)}) \;\;=\;\; G(\hat{x}) \;\;=:\;\; G_\infty \quad .$$

As it will turn out, there can be only one cluster point, which must also be a fixed point of the iteration (4.20). As a consequence of proposition 4.1 the sequence $\{x^{(j)}\}_{j\in\mathbb{N}}$ must therefore converge to $\hat{x} = x^*$. Let us commence our proof with the following technical proposition:

Proposition 4.2 *Let $\{x_j\}_{j\in\mathbb{N}}$ denote the sequence of vectors generated by the iteration (4.20). We assume, the iteration has been started at a feasible point $x_0 \in K$ with the cone of admissible vectors being defined by: $K := \{x \in \mathbb{R}^m \,|\, x \geq \psi\}$. If we introduce a functional $G : \mathbb{R}^m \longmapsto \mathbb{R}$ by:*

$$G(x) \quad := \quad x^T A\, x \;-\; 2\, f^T x$$

and assume $\hat{\Theta} \in (0, 1]$, we find that each iteration step decreases the "energy" $G(x^{(j+1)})$ of the approximate solution $x^{(j+1)} \in K$ in accordance with the estimate:

$$G(x^{(j+1)}) \;-\; G(x^{(j)}) \quad \leq \quad -\; \hat{\Theta}^2\; T_\Theta(x^{(j)})^T\, U_{\Theta\hat{\Theta}}\, T_\Theta(x^{(j)}) \quad . \tag{4.22}$$

If $D(S)$ denotes the diagonal part of the matrix S, the matrix $U_\vartheta \in \mathbb{R}^{m\times m}$ is hereby defined by:

$$U_\vartheta \quad := \quad \frac{1}{\vartheta}\, S \;+\; \frac{1}{\vartheta}\left(\frac{2}{\omega} \;-\; 1\right) D(S) \;-\; A \qquad ; \qquad \vartheta \;>\; 0 \quad .$$

Proof In the following we will use the notation we have introduced in (4.21). We note:

$$G(x^{(j+1)}) \;-\; G(x^{(j)}) \;\; = \;\; \hat{\Theta}^2\, y^T A\, y \;+\; 2\,\hat{\Theta}\, y^T (A\, x_j - f) \;\; = \;\; \hat{\Theta}^2\, y^T A\, y \;-\; 2\,\hat{\Theta}\,\Theta^{-1}\, y^T r \quad .$$

To evaluate the vector product $y^T r$ let us introduce another quadratic form $\hat{G} : \mathbb{R}^m \longmapsto \mathbb{R}$:

$$\hat{G}(z) \quad := \quad z^T S\, z \;-\; 2\, r^T z \quad .$$

The computation of the offset $y = T_\Theta(x^{(j)})$ proceeds through m stages starting with the initial vector $y^{(0)} = 0$ until the final offset $y^{(m)} = y$ is attained. If $e_i \in \mathbb{R}^m$ denotes the i-th Cartesian unit vector, we can state: $y^{(i)} - y^{(i-1)} = y_i\, e_i$ for any $i \in \{1, \ldots m\}$. Accordingly, we find:

$$\hat{G}(y^{(i)}) \;-\; \hat{G}(y^{(i-1)}) \;\; = \;\; S_{ii}\, y_i^2 \;+\; 2\, y_i \left(\sum_{j<i} S_{ij}\, y_j \;-\; r_i\right) \;\; = \;\; S_{ii}\, y_i^2 \;-\; 2\, y_i\, \hat{y}_i \quad .$$

We may now introduce a set of parameters $\omega_1, \ldots \omega_m$, such that

$$y_i \quad =: \quad \frac{\omega_i}{S_{ii}}\; \hat{y}_i \qquad ; \qquad i \;\in\; \{1, \ldots m\}$$

holds. With a view to the definition (4.21) we find, there are two possibilities: If y_i is equal to $\psi_i - x_i^{(j)}$, this quantity must be negative, since $x^{(j)}$ is admissible by assumption. We conclude:

$$0 \;\geq\; y_i \;=\; \frac{\omega_i}{S_{ii}}\; \hat{y}_i \;\geq\; \frac{\omega}{S_{ii}}\; \hat{y}_i \qquad \Longrightarrow \qquad 0 \;\leq\; \omega_i \;\leq\; \omega \quad .$$

The other possibility implies $\omega_i = \omega$ immediately. In the case $\omega_i = 0$ the component y_i vanishes. Since this component does not contribute to the energy functional $\hat{G}$, it may be ignored. Let us henceforth suppose, that $\omega_i \in (0, \omega]$ holds for every $i \in \{1, \ldots m\}$. We note:

$$\begin{aligned} y^T S\, y \;-\; 2 r^T y \quad &= \quad \hat{G}(y) \;-\; \hat{G}(0) \quad = \quad \sum_{i=1}^{m} \left\{\, \hat{G}(y^{(i)}) \;-\; \hat{G}(y^{(i-1)}) \,\right\} \\ &= \quad \sum_{i=1}^{m} \left\{\, S_{ii}\, y_i^2 \;-\; 2\,\frac{S_{ii}}{\omega_i}\, y_i^2 \,\right\} \quad \leq \quad \left(1 \;-\; \frac{2}{\omega}\right) y^T D(S)\, y \quad . \end{aligned}$$

Combining the above result with the very first equation presented in this proof we now obtain:

$$G(x^{(j+1)}) \;-\; G(x^{(j)}) \;\; = \;\; \hat{\Theta}^2\, y^T A\, y \;-\; \hat{\Theta}\,\Theta^{-1} \left\{\, y^T S\, y \;+\; \left(\frac{2}{\omega} \;-\; 1\right) y^T D(S)\, y \,\right\} \quad .$$

To finish our proof we have to show, that the iterate $x^{(j+1)}$ is again admissible. Since $\hat{\Theta} \leq 1$ and $x^{(j)} \geq \psi$ hold by assumption, we conclude:

$$x^{(j+1)} \quad \geq \quad x^{(j)} \;+\; \hat{\Theta}\,(\,\psi \;-\; x^{(j)}\,) \quad \geq \quad \psi \quad .$$

□

The quadratic form G we have introduced in the proof of proposition 4.2 is bounded from below. Hence, there are only two possibilities: If $x^{(j)}$ happens to coincide with the solution x^* of problem (4.19) for some index $j_0 \in \mathbb{N}$, the offset $T_\Theta(x^{(j_0)})$ vanishes. We find $x^{(l)} = x^*$ for each index $l > j_0$, so the sequence $\{x^{(j)}\}_{j\in\mathbb{N}}$ is convergent. If no such coincidence occurs, we observe $T(x^{(j)}) \neq 0$ for any index $j \in \mathbb{N}$. Let us suppose, that the product of the two damping factors is small enough to ensure, the matrix $U_{\Theta\hat{\Theta}}$ is positive definite. In this case the sequence $\{x^{(j)}\}_{j\in\mathbb{N}}$ minimises the functional G due to (4.22). We assume, that $\hat{x} \in \mathcal{G}$ is a cluster point, and we denote the smallest eigenvalue of the matrix $U_{\Theta\hat{\Theta}}$ by $u_0 > 0$. We note:

$$G(x^{(j)}) \;-\; G_\infty \quad \geq \quad \hat{\Theta}^2\, u_0 \sum_{l>j} \| T_\Theta(x^{(l)}) \|^2 \quad \geq \quad \hat{\Theta}^2\, u_0 \;\limsup_{l>j} \| T_\Theta(x^{(l)}) \|^2 \quad .$$

From the above inequality we may infer:

$$0 \quad = \quad \lim_{j\to\infty} G(x^{(j)}) \;-\; G_\infty \quad \geq \quad \hat{\Theta}^2\, u_0 \;\| T_\Theta(\hat{x}) \|^2 \quad \geq \quad 0 \quad .$$

Applying proposition 4.1 we conclude, that $\hat{x}$ is actually the solution x^* of problem (4.19). Since the sequence $\{x^{(j)}\}_{j\in\mathbb{N}}$ features exactly one cluster point, it is convergent.

Remark 4.5 The above proof relies on the fact, that the iteration (4.20) generates a sequence of admissible vectors, if the starting point is admissible itself. The projected SOR-scheme [55] produces a feasible point in its very first iteration, even if the starting vector has been infeasible. Our preconditioned defect correction scheme, however, will not necessarily yield an admissible vector $x^{(i+1)}$, if the preceding iterate $x^{(i)}$ has been infeasible. There are basically three ways to remedy this seeming drawback of our method: we can precede the iteration with a projection into the cone of admissible vectors and start with a feasible approximation; we can fix $\hat{\Theta} = 1$ or we can modify the projection to account for the damping:

$$y_i \quad := \quad \max\Big\{ \frac{\psi_i - x_i}{\hat{\Theta}}\,,\; \frac{\omega}{S_{ii}}\; \hat{y}_i \Big\} \quad .$$

Both the proof of proposition 4.1 and of proposition 4.2 carry over with only minor modifications. The scaling of the shifted obstacle ensures $x + T_\Theta(x) \in K$ for any choice $x \in \mathbb{R}^m$.

4.3.6 Avoiding the global complementary condition

Eliminating λ from (4.12) has the advantage, that the resulting complementary problem for μ involves simple box constraints and allows for a discretisation, that facilitates the use of nested trial spaces and the straightforward implementation of a multilevel solver. The drawback of this procedure is the necessity to solve a global linear problem in each smoothing step. However, for any fixed $\lambda \in \mathbb{R}^{m_3}$ the Lagrange multiplier for the admissibility constraint can be identified as the solution of the following linear complementary problem:

$$\begin{aligned} DM^{-1}D^T\mu \quad &\geq \quad DM^{-1}(\,t \,+\, F^T\lambda\,) \;-\; d \quad =: \quad d(\lambda) \\ \wedge \qquad \mu \;\geq\; \psi \quad &\wedge \quad (\,\mu - \psi\,)^T\big(\, DM^{-1}D^T\,\mu \;-\; d(\lambda)\,\big) \quad = \quad 0 \quad . \end{aligned} \tag{4.23}$$

As the matrix $DM^{-1}D^T$ is a block-diagonal, formula (4.23) actually describes numerous small, independent complementary problems, which correspond to the individual elements compounding the mesh $\mathcal{M}_h$. Let us denote by $\Lambda : \mathbb{R}^{m_3} \longrightarrow \mathbb{R}^{m_2}$ the very map, which assigns to each vector $\lambda \in \mathbb{R}^{m_3}$ the corresponding solution of problem (4.23). From (4.12) a nonlinear equation for the Lagrange multiplier λ is obtained:

$$T(\lambda) \quad := \quad FM^{-1}F^T\,\lambda \;-\; FM^{-1}D^T\,\Lambda(\lambda) \quad = \quad -\,FM^{-1}\,t \quad . \tag{4.24}$$

In order to solve this equation by a FAS variant [28] of the multigrid method, it is necessary to specify a smoothing iteration and suitable transfer mechanisms, that map the defect and the approximate solution from a coarser grid $\mathcal{I}_{h_l}$ to the next finer one $\mathcal{I}_{h_{l+1}}$ and back. Unfortunately, the trial spaces $P^k(\mathcal{I}_{h_0}), P^k(\mathcal{I}_{h_1}), \ldots$ are not nested and canonical injection or restriction operators

are not available. However, the Lagrange multipliers of the normal fluxes approximate the values of the displacement on the element interfaces [9]. Hence, the injection and restriction operators can be defined by treating the components of the vector λ as associated with a Crouzeix-Raviart ansatz [54] for the displacement variable. Consequently, the level transfer techniques described in [25] may be used. A simpler alternative is suggested in [31]. In either case the choice of proper damping parameters is critical to the success of the resulting multilevel scheme. Numerical experiments indicate that a relaxation of the fine grid contribution to the right hand side on the coarser grid may be the most successful strategy:

$$f_{l-1} \;:=\; T_{l-1}(\, R_{l-1}(\lambda_l^{(j)})\,) \;+\; \vartheta\, I_{l-1}^T(\, f_l \,-\, T_l(\lambda_l^{(j)})\,) \quad .$$

Hereby, the effect of smoothing has been ignored for the sake of simplicity. The injections $I_0, \ldots I_{L-1}$ are defined by (4.17), while the restrictions $R_0, \ldots R_{L-1}$ have proved most effective, if they are constructed in the spirit of (4.18). The relaxation parameter is denoted by $\vartheta > 0$. Its impact on the coarse grid problem can be compensated, when the coarse grid correction is injected into the finer grid:

$$\lambda_l^{(j+1)} \;:=\; \lambda_l^{(j)} \;+\; \vartheta^{-1}\, I_{l-1}(\, \lambda_{l-1}^{(j+1)} \,-\, R_{l-1}(\lambda_l^{(j)})\,) \quad .$$

Again, the effect of smoothing has been ignored. If the obstacle ψ were sufficiently far removed from the solution u^* of the problem $\mathbf{P}$, the Lagrange multiplier μ would stay strictly positive and the complementary problem (4.23) would be reduced to an equation. Consequently, (4.24) would turn into a linear system involving the matrix

$$T \;:=\; FM^{-1}F^T \,-\, FM^{-1}D^T(DM^{-1}D^T)^{-1}DM^{-1}F^T \quad .$$

This matrix has no more than $n+1$ blocks per row, whose size only depends on the order of the ansatz for the Lagrange multiplier λ. As T can be computed and stored with optimal numerical complexity, a fixed point iteration of the form

$$\lambda^{(j+1)} \;:=\; \lambda^{(j)} \,-\, \hat{T}^{-1}FM^{-1}\left(\, t + F^T\lambda^{(j)} - D^T\Lambda(\lambda^{(j)})\,\right)$$

may be utilised as a smoother, whereby the preconditioner $\hat{T}$ can be based on a regular splitting of T: among the possible smoothing iterations are the damped Gauß-Seidel or the SOR scheme. Due to the special structure of the auxiliary problem (4.23) the overall numerical complexity of one smoothing step will not exceed $\mathcal{O}(m_3)$.

Chapter 5

Numerical Experiments

Below we are going to assess the usefulness of those hypercycle estimates, we have discussed in chapter 2. We will look at a number of generic test cases, instances of the Laplace and the obstacle problem, in order to compare various methods how to construct the constitutive parameter $\sigma_h \in L^2(\Omega, \mathbb{R}^n)$. We have seen in the sections 2.1.4 and 2.2.7 respectively, that said vector field should be an approximation to the solution of a perturbed variational problem posed in its dual formulation. Several approaches to finding a good hypercycle estimate may be inferred from this fact: 1.) either the dual formulation of the original variational problem or the dual formulation of the perturbed problem is solved numerically using an adequate discretisation scheme, 2.) the numerical solution of the primal formulation or its perturbed counterpart is post-processed to obtain a sufficiently regular approximation of the dual variable, 3.) a minimisation algorithm is employed to improve the error bound with respect to the dual parameter $\sigma_h \in L^2(\Omega, \mathbb{R}^n)$.

To fix some sort of standard against which we can compare our findings a conventional a posteriori error estimator has been implemented for the Laplace problem. The estimator is based on the computation of residual expressions on an element by element basis and therefore must rely on the best approximation property to be met by the numerical solution. Since an equivalent estimator is not available for the obstacle problem, we have followed the ideas presented in [86] and written a direct a posteriori estimator for the following variational statement:

$$\langle \nabla x_\epsilon, \nabla v \rangle_\Omega \; + \; (\epsilon^{-1} \min\{0, x_\epsilon - \psi\}, v)_\Omega \;\; = \;\; (f, v)_\Omega \quad ; \quad v \in H_0^1(\Omega)$$

with $x_\epsilon \in H_0^1(\Omega)$ denoting the solution of the relaxed obstacle problem and $\epsilon \in L^\infty(\Omega)$ a suitable penalisation of the feasibility constraint $x_0 \in V_\psi$. Reliable energy error bounds for the obstacle problem are obtained from a more or less conventional a posteriori error estimator for the above statement and an a priori estimate for the consistency error we incur due to the regularisation. Controlling this latter contribution may be achieved by means of the function ϵ.

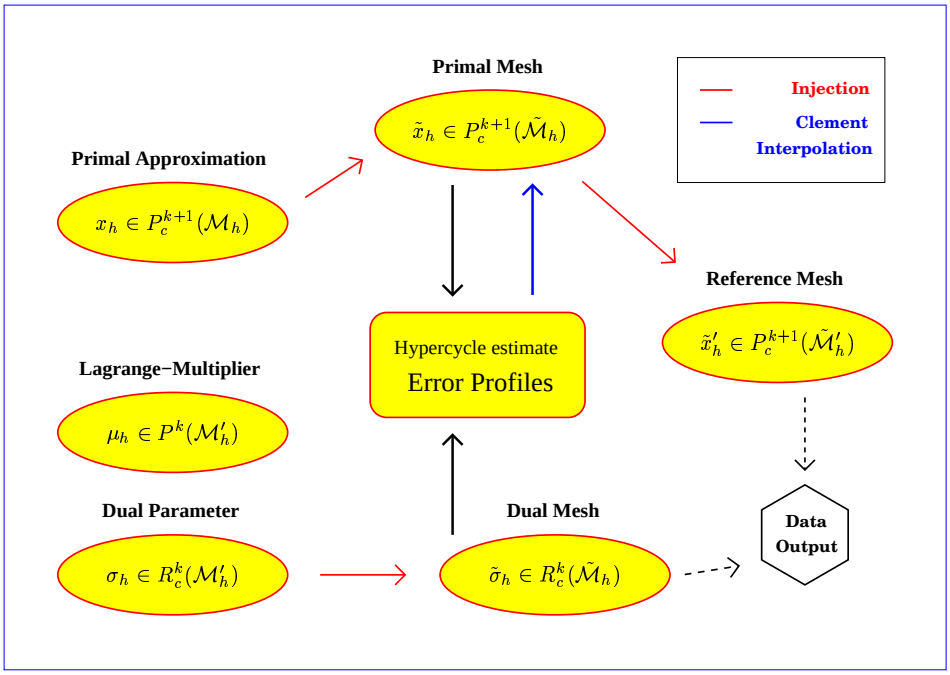
5.1 General remarks on the simulation code

The concept of an useful a posteriori error estimator is perhaps not as clearly defined as it may seem at first. The computation of an error bound for some approximate solution to a variational problem is a sensible undertaking only in so far, as we are able and willing to improve our numerical results in accordance with the information provided by the error estimator. The knowledge of a reliable bound on the error may certainly be helpful when we have to decide whether we can accept the numerical solution at hand. However, if we proceed to globally refine our mesh, as we have found the accuracy of the current approximation to be insufficient, we practically waste the computational effort, we have spent on the error estimate. If a global mesh refinement is our most appropriate response, we should have estimated the proper mesh width prior to any computations with the help of an a priori estimate.

Accepting such a premise, we come to the conclusion, that an useful error estimator should provide us with some sort of profile, we can assess in order to determine regions of the mesh, we need to refine locally. The involved algorithms must be able to handle the emerging meshes: unstructured and locally refined formations that may contain various types of finite elements or even agglomerates of elements, which have been formed to avoid the so called *hanging nodes*. If we employ finite dimensional trial spaces to represent the dual parameters which are necessary to evaluate generalised hypercycle estimates, we can carry this notion one step further. We may adapt not only our meshes with the help of which we seek a possibly accurate numerical solution of the primal formulation, we may also want to adapt the meshes on which we construct our dual parameters respectively solve the dual problem. Hence, our finite element machinery should not only indicate, which elements to refine in the "primal" mesh, but in the *"dual"* one as well.

The technical difficulties we must address to facilitate this much flexibility are substantial. Arguably, the most challenging part consists in generating a mesh, such that we can construct two trial spaces into the first of which we can inject the dual parameter, while the second one accommodates the numerical solution of the primal formulation. After such a joint mesh has been formed in some kind of merging procedure, the hypercycle estimate can be evaluated by visiting each element in turn and assembling the element contributions for all relevant types of shape functions. In section 4.2 we have endeavoured to detail some of our technical solutions: rather than dealing with several types of shape functions on the same mesh, we create one mesh for each type of finite element we need in our computations. After sorting the elements according to certain geometrical properties we can assess the error estimates by sweeping the meshes' top layers synchronously. In consequence, a fully adaptive finite element code for e. g. the Laplace problem will involve up to five different meshes: one mesh for the numerical solution of the primal formulation, two meshes for the computation of the dual formulation and two corresponding meshes, into which we inject both the primal approximation and the dual parameter.

Figure 5.1: Outline of the code's structure



The relevant numerical quantities having been computed by sweeping those meshes we can compare the results against the true energy error and its profile. If an analytic solution is not available to us, we need one further mesh on which we can compute the reference solution. For simplicity, we also use this very mesh to output the data in an external format suitable for further processing e. g. as an *unstructured cell data* file. Accordingly, we have to transfer data from various meshes to the finest one - converting the information, if necessary, into vertex oriented data. The profiles predicted by the computable hypercycle estimates (2.8) and (2.16) respectively (2.19) are the very quantities, the elements contribute to the a posteriori error bound. As integrals over single elements these quantities are per se cell oriented. To transform the information into vertex oriented data, we compute weighted averages using continuous shape functions $\psi_j \in P_c^1(\mathcal{M}_h)$ of Lagrange type, which correspond to the vertices $\xi_j \in \Omega$ of the mesh $\mathcal{M}_h$. Thereby, the interpolation operator $\Pi_1 : L^2(\Omega) \longrightarrow P_c^1(\mathcal{M}_h)$ is defined by:

$$\Pi_1 u \quad := \quad \sum_{M \in \mathcal{M}_h} \sum_{\xi_j \in \operatorname{clo} M} \frac{\psi_j}{|\operatorname{supp} \psi_j|} \int_M u \; d\xi \quad .$$

The data transfer of vertex oriented information from one of the coarser meshes to the finest one is comparatively simple, as the trial spaces for both the primal and the dual formulation are nested. For instance, the numerical solution $x_h \in P_c^k(\mathcal{M}_h)$ is mapped into the trial space $P^k(\tilde{\mathcal{M}}_h)$

first. The image $I_k x_h$ is then injected into the trial space $P^k(\tilde{\mathcal{M}}'_h)$. The class `FemGrid` contains a method, which applies the interpolation operator Π_1 to the resulting function. The information pertaining to the dual formulation need not be compared against the reference solution. Hence, it is not necessary to increase the amount of data to be exported by mapping e. g. the dual parameter into the trial space $P^k(\tilde{\mathcal{M}}_h)$. Instead, the Clément interpolation Π_1 is applied to each component of the field $\sigma_h \in R^k(\tilde{\mathcal{M}}'_h)$ directly and the results are stored in a separate file.

5.2 Energy Error Estimates for the Laplace Problem

In the following paragraphs we present a number of numerical experiments which are designed to explore those features of the hypercycle estimates we have failed to analyse properly in the section 3.3. We will numerically solve the Dirichlet problem using the simplest possible conforming ansatz of Lagrange type to construct our approximate solution. The approximation error is bound in the energy norm with the help of a conventional a posteriori estimator based on the evaluation of local residuals. Thus we can provide a baseline, against which the subsequent results may be compared. In the second section we compute hypercycle estimates by smoothing the gradient of our numerical solution, such that the resulting fields are contained in the space $H_{\text{div}}(\Omega)$. Section 5.2.3 will deal with error majorants, that are obtained by solving the dual formulation of the Dirichlet problem. In the last section we will investigate the possibility of improving the error bounds by minimising the hypercycle estimates in terms of the dual parameter σ_h.

5.2.1 A description of the experiments

Three model problems shall be considered, whose analytical solutions feature different degrees of smoothness. We will solve all three problems on a circular disc of radius 1 imposing homogeneous Dirichlet boundary conditions. The forcing functions will be defined in polar coordinates:

Example A:

$$f_1(r) \quad := \quad \begin{cases} f_0 & ; \quad 0 \le r \le r_0 \\ 0 & ; \quad \text{else} \end{cases} \quad .$$

Our actual choices for the two parameters in the above definition read: $f_0 = -4$ and $r_0 = 0.25$. Due to the radial symmetry of the load f_1, the analytical solution $x \in H^2(\Omega)$ of the Dirichlet problem can be computed by considering the following two boundary value problems:

$$-\frac{1}{r}\frac{\partial}{\partial r}\left(r\,\frac{\partial}{\partial r}u\right) \quad = \quad \begin{cases} f_0 & ; \quad \text{on } (0, r_0) \\ 0 & ; \quad \text{on } (r_0, 1) \end{cases} \quad .$$

There are four boundary conditions we have to meet. If we denote the solution of the first differential equation on the interval $[0, r_0]$ by u_0 and the solution of the other equation by u_1, we can obtain immediately: $u_1(1) = 0$ and $u_0'(0) = 0$, since the analytical solution x_0 must be continuous and therefore bounded. The continuity of x implies moreover: $u_0(r_0) = u_1(r_0)$. The fourth requirement stems from the fact, that the solution x is indeed continuously differentiable as well. Hence, we must impose: $u_0'(r_0) = u_1'(r_0)$. Using the two latter constraints we can fix the remaining parameters in the ansatz below:

$$x(r, \phi) \quad = \quad \begin{cases} \zeta_0 - f_0/4\, r^2 & ; \quad 0 \le r \le r_0 \\ \zeta_1 \ln(r) & ; \quad \text{else} \end{cases} \quad .$$

By simple algebraic manipulations we find eventually:

$$\zeta_0 \quad = \quad \frac{f_0}{4}\, r_0^2 \left(1 - \ln\left(r_0^2\right)\right) \qquad ; \qquad \zeta_1 \quad = \quad -\frac{f_0}{2}\, r_0^2 \quad .$$

Since the load function f_1 is discontinuous but polynomial within both intervals $[0, r_0)$ and $(r_0, 1]$, in cannot be contained in $H^1(\Omega)$. Consequently, the analytical solution x of the Dirichlet problem is an element of the space $H^2(\Omega)$, but is not contained in $H^3(\Omega)$.

Example B:

$$f_2(r) \quad := \quad \begin{cases} f_0 & ; \quad 0 \le r \le r_1 \\ f_0 \, \dfrac{r - r_2}{r_1 - r_2} & ; \quad r_1 \le r \le r_2 \\ 0 & ; \quad \text{else} \end{cases} \quad .$$

We can apply the very same methodology we have just outlined to find the analytical solution of the Dirichlet problem, which corresponds to the above load function. Our ansatz $u \in C^{1,1}([0,1])$ for the radial dependency of the solution contains 4 parameters, which we cannot fix a priori with a view to the boundary conditions $u_0'(0) = 0$ and $u_2(1) = 0$. We find:

$$x(r,\phi) \quad = \quad \begin{cases} \zeta_0 \, - \, f_0/4 \, r^2 & ; \quad 0 \le r \le r_1 \\ \zeta_1 \, + \, \zeta_2 \, \ln(r) \, - \, (\, \zeta_3 + \zeta_4 \, r \,) \, r^2 & ; \quad r_1 \le r \le r_2 \\ \zeta_5 \, \ln(r) & ; \quad \text{else} \end{cases} \quad .$$

At the points r_1 and r_2 the following interface conditions must be met to warrant the required regularity of the analytical solution: $u_{i-1}(r_i) = u_i(r_i)$, $u_{i-1}'(r_i) = u_i'(r_i)$ with $i \in \{1, 2\}$. Hereby, as in the previous example, the symbol u_0 designates the restriction of the ansatz u onto the first interval $(0, r_1)$, the symbol u_1 designates the restriction of u into the interval (r_1, r_2) and u_2 denotes the restriction of u onto the third interval $(r_2, 1)$. The remaining parameters read:

$$\zeta_3 \quad := \quad \frac{f_0 \, r_2}{4 \, (r_2 - r_1)} \qquad ; \qquad \zeta_4 \quad := \quad \frac{f_0}{9 \, (r_1 - r_2)} \quad .$$

The algebraic manipulations necessary to compute ζ_0 through ζ_2 and ζ_5 are cumbersome but straightforward. Therefore, they shall not be detailed here. We note, that the analytical solution x is contained but in $H^3(\Omega)$, since the load f_2 is not smooth enough: $f \notin H^2(\Omega)$.

Example C:

$$f_3(r) \quad := \quad e^{-\frac{\alpha}{r}} \left\{ \left(1 \, - \, \frac{1}{r} \right) \frac{\alpha^2}{r^4} \, + \, \frac{3 \, \alpha}{r^4} \, - \, \frac{\alpha + 1}{r^3} \right\} \quad ; \quad r \ge 0 \ .$$

The last example is intended to provide a smooth solution, which can nevertheless prove difficult to approximate by a finite elements scheme due to its steep gradients and a peculiar behaviour in some neighbourhood around the centre of the computational domain. The load f_3 has been obtained simply by applying the Laplace operator to the targeted solution:

$$x(r,\phi) \quad = \quad e^{-\frac{\alpha}{r}} \left(1 \, - \, \frac{1}{r} \right) \quad ; \quad r \ge 0 \ .$$

The parameter $\alpha > 0$ controls the character of the function x. In the limit $\alpha \to 0$, the analytical solution features a pole. Therefore, the finite dimensional problem will have an "almost singular" solution, if α is but small enough. The numerical experiments are carried out with $\alpha = 3/4$.

5.2.2 On the choice of certain constants

Unfortunately, both the evaluation of hypercycle estimates and the computation of conventional a posteriori error estimators based on element residuals require our knowledge of quantities, which we can but estimate in most situations. In the first case, we have to determine a positive lower bound for the constant λ_0 which we have introduced in (1.4) to describe the coercivity of the operator Λ. In the latter case we must find two numbers: a stability constant $L_0 > 0$, which is related to the continuity of the operator $L := (\Lambda^* \Lambda)^{-1}$:

$$\| \, L \, x^* \, \|_X \quad \le \quad L_0 \, \| \, x^* \|_{X^*} \qquad ; \qquad x \in X \quad ,$$

and an interpolation constant $C > 0$ describing the accuracy of the finite element interpolation operator $I_h : H^1(\Omega) \longrightarrow P_c^k(\mathcal{M}_h)$. Let h_M denote the diameter of some element $M \in \mathcal{M}_h$. We may define the sharpest possible interpolation constant on that particular element by:

$$C_M \ := \ \sup_{v \in H_0^1(\Omega)} \frac{\| \, v - I_h v \, \|_M}{h_m \, | \, v \, |_{M,1}} \qquad \text{resp.:} \qquad \tilde{C}_M \ := \ \sup_{v \in H_0^1(\Omega)} \frac{\| \, v - I_h v \, \|_{\partial M}}{\sqrt{h_m} \, | \, v \, |_{M,1}} \quad .$$

If these number were known for each element in the mesh $\mathcal{M}_h$, the global interpolation constant C would be computable in accordance with section 3.3.1 as:

$$C \;:=\; \sup_{v\in H_0^1(\Omega)} \frac{1}{|\,v\,|_{\Omega,1}} \Big\{ \sum_{M\in\mathcal{M}_h} \big(\, C_M^2 \,+\, \tilde{C}_M^2 \,\big)\, |\,v\,|_{M,1}^2 \Big\}^{1/2} \;\leq\; \max_{M\in\mathcal{M}_h} \sqrt{C_M^2 \,+\, \tilde{C}_M^2} \quad .$$

To all practical purposes, the above expression is impossible to determine exactly. Moreover, any modification to the mesh $\mathcal{M}_h$ will affect the interpolation constant. Hence, it would be necessary to reassess the approximation properties of the finite element space several times during the course of any finite element computation, which relies on local mesh refinements to reduce the approximation error. Since the residual based error estimator is known to be asymptotically exact, we may hope, to provide sharp estimates for C simply by evaluating the a posteriori error bounds and tuning the constants accordingly. However, such an approach may require us to solve our variational problem (or alternatively any other problem, whose stability constant is known to us) on increasingly finer meshes to a higher accuracy than may really be necessary. For if we do not carry this refinement procedure far enough, we can never rule out the possibility of underestimating the interpolation constant.

The Poincaré constant, respectively its inverse λ_0, as defined by (1.4), may be difficult to compute for arbitrary domains $\Omega \subset \mathbb{R}^n$. In the case of a circular disc with radius 1 the calculations are extremely simple, however. Using polar coordinates we can express the Euclidean norm of the gradient ∇v of some smooth function $v \in C_0^\infty(\Omega)$ by:

$$\|\,\nabla v\,\|_{\mathbb{R}^2}^2 \quad = \quad \left|\,\frac{\partial v}{\partial r}\,\right|^2 \;+\; \frac{1}{r^2}\left|\,\frac{\partial v}{\partial \phi}\,\right|^2 \quad \geq \quad \left|\,\frac{\partial v}{\partial r}\,\right|^2 \quad .$$

We infer, that it is sufficient to introduce the radial derivative into the integral, which represents the L^2-norm of v. In fact, we may exploit the identity:

$$v(r,\phi) \quad = \quad -\int_r^1 \frac{\partial v}{\partial r}(\rho,\phi)\;\; d\,\rho \qquad ; \qquad r\in[0,1]$$

and bound the square of the norm $\|\,v\,\|_\Omega$ with the help of Hölder's inequality:

$$\begin{aligned}
\int_0^{2\pi}\!\!\int_0^1 r\,\Big|\int_r^1 \frac{\partial v}{\partial r}\,d\,\rho\,\Big|^2\;d\,r\,d\,\phi \quad &\leq \quad \int_0^{2\pi}\!\!\int_0^1 \Big\{\, r\int_r^1 \rho\,\Big|\,\frac{\partial v}{\partial r}\,\Big|^2 d\,\rho \int_r^1 \frac{d\,\rho}{\rho}\,\Big\}\;d\,r\,d\,\phi \\
&\leq \quad \Big\{\int_0^{2\pi}\!\!\int_0^1 \rho\,\Big|\,\frac{\partial v}{\partial r}\,\Big|^2 d\,\rho\,d\,\phi\,\Big\}\Big\{\int_0^1 r\int_r^1 \frac{d\,\rho}{\rho}\;\;d\,r\,\Big\} \\
&\leq \quad -\,|\,v\,|_{\Omega,1}^2 \int_0^1 r\,\ln(r)\;\,d\,r \quad = \quad \frac{1}{4}\,|\,v\,|_{\Omega,1}^2 \quad .
\end{aligned}$$

By density we can extend the above result to any function $v \in H_0^1(\Omega)$. We conclude, that we may bound the parameter λ_0 from below by 2.

Remark 5.1 In the present context the stability constant L_0 reads exactly 1. The interpolation constant C will depend on the mesh depicted in figure 5.2. Since the mesh is refined but uniformly with each triangle being split into four self-similar elements, the mesh adaption will not affect this latter quantity. We have chosen $C = 1.0$, though the example 3 in [49] demonstrates, the local interpolation constants C_M alone can become as large as 1.5. Our choice is prompted by the possibility, that the estimate in [49] may be overly pessimistic for a substantial number of elements, and the consideration, that many people will start their finite element computations in default of more accurate data with just this value to scale the element residuals.

5.2.3 Error estimates on uniformly refined meshes

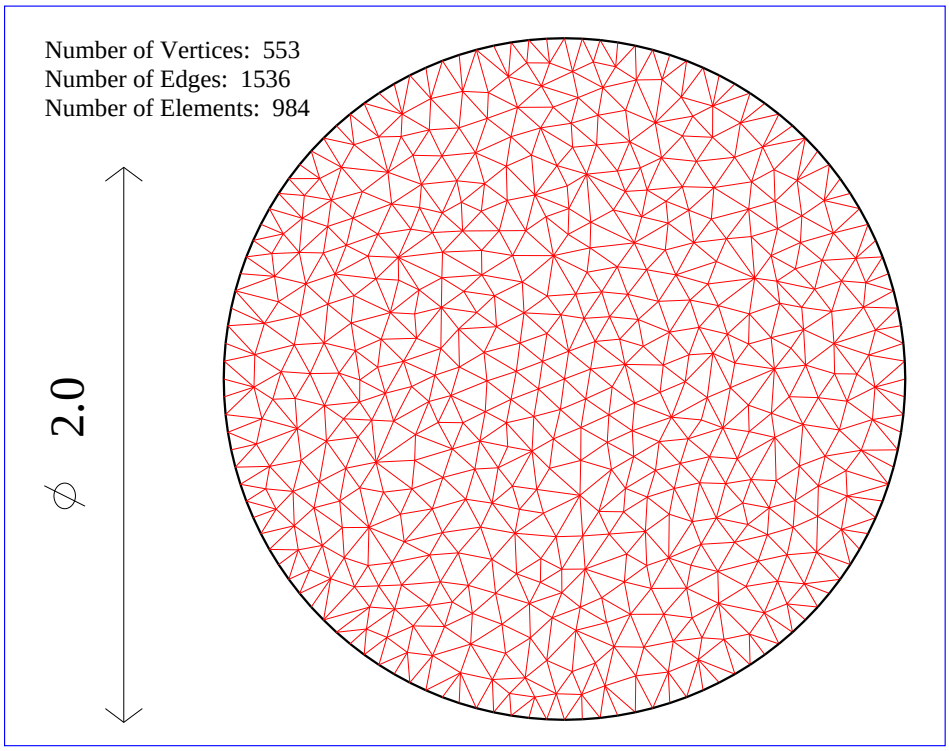
Having introduced those problem, which shall constitute our test bed for the hypercycle estimates we have derived in chapter 1, let us proceed by comparing the performance of various a posteriori estimates for the approximation error under uniform mesh refinement. The finite element scheme is implemented on meshes, which have been obtained from the very mesh depicted in figure 5.2 by successive one on four refinement of all elements in the top layer as illustrated in figure 4.5. The coarsest mesh has been created with an external tool, called *Triangle* [132], which creates quality conforming Delauney triangulation for basically any domain, whose boundary may be described by a planar straight line graph. Our results have been collected in appendix C. Throughout that section we use the following notation: $x_h \in P_c^1(\mathcal{M}_h)$ denotes an approximation to the analytical solution $x_0 \in H_0^1(\Omega)$ of the primal formulation. The vector fields $\rho_0 \in R_c^0(\mathcal{M}_h)$, $\rho_1 \in R_c^1(\mathcal{M}_h)$ and $\sigma_1 \in B_c^1(\mathcal{M}_h)$ designate numerical solutions of the corresponding dual formulation. Additionally, the vector fields $\hat{\rho}_0 \in R_c^0(\mathcal{M}_h)$, $\hat{\rho}_1 \in R_c^1(\mathcal{M}_h)$ and $\hat{\sigma}_1 \in B_c^1(\mathcal{M}_h)$ are obtained from:

$$\hat{\tau}_h \in Y_h \quad : \quad \langle\, \hat{\tau}_h, \tau_h \,\rangle_\Omega \;=\; -\,(\, x_h, \nabla\cdot \tau_h \,)_\Omega \quad ; \quad \tau_h \in Y_h \tag{5.1}$$

with Y_h denoting one of the three vector valued trial spaces mentioned above. The quantity $\mathcal{H}^*(x_h, \tau)$ designates a generalised hypercycle estimate for the numerical approximation x_h, which is evaluated at the point $\tau \in H_{\text{div}}(\Omega)$. The asterisk indicates, that this estimate is optimal with respect to the equilibration parameter κ. If the vector field $\tau \in (P_c^1(\mathcal{M}_h))^2$ has been obtained from the gradient ∇x_h of the numerical solution with the help of the averaging operator Π introduced in section 3.3.1, we abbreviate the generalised hypercycle estimate by $\mathcal{H}^*(x_h)$. Again, the equilibration parameter κ is chosen in such a way, that the resulting error bound is minimal. Each of the hypercycle estimates consists of the following two components:

$$M_D^{(\kappa)}(x_h, \tau) \;:=\; \frac{1+\kappa}{\kappa}\, \|\, \nabla x_h - \tau \,\|_\Omega^2 \quad ; \quad M_R^{(\kappa)}(x_h, \tau) \;:=\; \frac{1+\kappa}{\lambda_0^2}\, \|\, \nabla\cdot \tau + f \,\|_\Omega^2 \;\; .$$

Figure 5.2: The coarsest mesh



The asterisk indicates, whether the optimal equilibration parameter κ^* has been selected. This

quantity can be computed for any initial choice of the parameter $\kappa > 0$ by the formula:

$$\kappa^* \quad := \quad \sqrt{\frac{\kappa\, M_D^{(\kappa)}(x_h,\tau)}{M_R^{(\kappa)}(x_h,\tau)}} \qquad . \tag{5.2}$$

By $\mathcal{R}_h(x_h)$ we denote the conventional a posteriori error estimator discussed in section 3.3.1, which is based on element residuals. The ratio between some estimate of the approximation error and its true value is usually termed the *efficiency index* of the error estimator. Provided the error bound is reliable, this quantity is larger or equal 1:

$$I_{\text{eff}} \quad := \quad \frac{\sqrt{\mathcal{H}^{(\kappa)}(x_h,\tau)}}{|\, x_h - x_0 \,|_{\Omega,1}} \qquad .$$

In the following we will not trace the dependence of I_{eff} either on the character of the error estimate employed or on our choice of the dual parameter τ.

Figure 5.3: Efficiency Indices I_{eff} for various Error Estimates

Estimate	Example	Level 0	Level 1	Level 2	Level 3	Level 4
$\mathcal{R}_h(x_h)$	A	4.24	4.39	4.44	4.47	4.50
$\mathcal{H}^*(x_h)$	A	9.53	13.95	19.43	27.29	38.35
$\mathcal{H}^*(x_h,\rho_0)$	A	7.24	10.28	13.89	19.84	27.24
$\mathcal{H}^*(x_h,\hat{\rho}_0)$	A	13.23	18.97	26.36	36.68	51.18
$\mathcal{H}^*(x_h,\sigma_1)$	A	6.95	9.99	13.59	19.54	26.94
$\mathcal{H}^*(x_h,\hat{\sigma}_1)$	A	12.25	17.45	23.98	33.41	46.75
$\mathcal{H}^*(x_h,\rho_1)$	A	5.01	6.52	8.63	12.23	16.71
$\mathcal{H}^*(x_h,\hat{\rho}_1)$	A	25.15	51.90	106.03	214.55	433.93
$\mathcal{R}_h(x_h)$	B	4.49	4.52	4.56	4.59	4.59
$\mathcal{H}^*(x_h)$	B	5.49	8.72	11.61	15.74	21.6
$\mathcal{H}^*(x_h,\rho_0)$	B	3.00	2.98	2.97	2.97	2.97
$\mathcal{H}^*(x_h,\hat{\rho}_0)$	B	10.91	15.18	20.23	27.22	37.21
$\mathcal{H}^*(x_h,\sigma_1)$	B	2.75	2.73	2.73	2.73	2.73
$\mathcal{H}^*(x_h,\hat{\sigma}_1)$	B	10.04	14.31	19.14	26.16	36.16
$\mathcal{H}^*(x_h,\rho_1)$	B	1.21	1.15	1.10	1.07	1.05
$\mathcal{H}^*(x_h,\hat{\rho}_1)$	B	24.23	49.66	101.81	206.69	415.97
$\mathcal{R}_h(x_h)$	C	4.16	4.37	4.45	4.48	4.49
$\mathcal{H}^*(x_h)$	C	10.05	10.99	12.28	15.01	19.52
$\mathcal{H}^*(x_h,\rho_0)$	C	6.43	7.12	7.18	7.19	7.19
$\mathcal{H}^*(x_h,\hat{\rho}_0)$	C	12.95	17.17	21.21	26.21	33.50
$\mathcal{H}^*(x_h,\sigma_1)$	C	6.16	6.84	6.90	6.91	6.91
$\mathcal{H}^*(x_h,\hat{\sigma}_1)$	C	11.67	15.31	18.92	24.42	32.66
$\mathcal{H}^*(x_h,\rho_1)$	C	2.75	1.89	1.43	1.22	1.11
$\mathcal{H}^*(x_h,\hat{\rho}_1)$	C	24.24	52.19	108.74	221.11	445.21

Our findings are summarised in figure 5.3. The numerical results corroborate the deliberations, we have put forth in section 3.3.1, that the efficiency index of the hypercycle estimate (2.8) cannot be bounded uniformly in the mesh parameter h, unless the data of the problem has some surplus smoothness. The degradation of the error bound is caused by the introduction of Friederich's inequality in order to control the residual in the first duality relation. Consequently, the second contribution $M_R^{(\kappa)}(x_h,\cdot)$ to the hypercycle estimate $\mathcal{H}^{(\kappa)}(x_h,\cdot)$ is lacking two powers of the mesh

parameters h if compared to the first contribution $M_D^{(\kappa)}(x_h,\cdot)$. To some extent, this imbalance is compensated for by the equilibration mechanism implemented through (5.2).

Projecting the gradient of the primal approximation into a conforming trial space appropriate for the numerical solution of the dual formulation, fails to produce satisfactory error estimates, even if the analytical solution is sufficiently smooth. To a somewhat lesser extent this also holds true for the simple gradient recovery scheme we have discussed in section 3.3.1. A closer look at the tables in appendix C reveals, that the disappointing results are again caused by the lacking convergence of the residual expression associated with the first duality relation. Interestingly, the use of trial spaces with more degrees of freedom to satisfy the admissibility condition $\nabla\cdot\sigma+f=0$ does not improve the resulting hypercycle estimates. On the contrary, projecting the gradient ∇x_h of the primal approximation into Raviart-Thomas spaces of higher order proves futile!

Provided the data is smooth enough and the dual parameter has been obtained by solving the dual formulation of the Laplace problem numerically, the efficiency indices of the various hypercycle estimates stay uniformly bounded. Moreover, if a Raviart-Thomas space of higher order is employed and if the load function is contained at least in $H^2(\Omega)$, the resulting error bound even seems to be asymptotically exact. Our results for the example B may be too optimistic, as in this particular case the load function is contained in the image of the ansatz $R_c^1(\mathcal{M}_h)$ under the action of the divergence operator but for two concentric annuli, whose joint area shrinks in proportion to the mesh parameter h.

5.2.4 Minimising the generalised hypercycle estimates

The unsatisfactory results of exploiting the mapping, which connects the solution of the primal formulation with that of its dual counterpart, prompt us to look for other inexpensive methods of computing sharp hypercycle estimates. On such method may consist in applying an optimisation algorithm to improve an existing error bound. Since the hypercycle estimate $\mathcal{H}^{(\kappa)}(x_h,\cdot)$ is a convex functional, we may for instance use a line search procedure to relax each degree of freedom in turn, associated with the dual parameter σ_h. Actually, our numerical tests rely on a line search algorithm, which is based on the successive employment of quadratic interpolation and the minimisation of the interpolant. The algorithm is known under the denomination *Bracketing*. An exhaustive description of its functionality can be found e. g. in [140].

As the evaluation of the error bound $\mathcal{H}^{(\kappa)}(x_h,\cdot)$ requires a complete sweep across the top layer of the mesh $\mathcal{M}_h$, the computational effort would be prohibitive, if we implemented the line search procedure in a straightforward fashion. However, we may decompose the hypercycle estimate into sums of local contributions, such that the line searches will affect only one of these contributions at a time. Let $\psi_i\in P_c^1(\mathcal{M}_h)$ denote the shape function associated with the vertex $\xi_i\in\Omega$ respectively the degree of freedom indexed i. By $\Omega_i\subset\Omega$ let us designate the support of said shape function. We introduce the quantities:

$$\mathcal{H}_i^{(\kappa)}(x_h,\tau) \quad := \quad \frac{\kappa+1}{\kappa}\int\limits_{\Omega_i}\Big\{\,(\,\nabla x_h-\tau\,)^2\;+\;\frac{\kappa}{\lambda_0^2}\,(\,\nabla\cdot\tau+f\,)^2\,\Big\}\,d\xi \qquad ; \qquad \tau\in H_{\mathrm{div}}(\Omega)$$

for any degree of freedom, that is featured by the ansatz $P_c^1(\mathcal{M}_h)$. Let us suppose, that the dual parameter τ_h be contained in the trial space $(P_c^1(\mathcal{M}_h))^2$. We may compute the update of the objective function $\mathcal{H}^{(\kappa)}(x_h,\cdot)$ along some vector $\tau\,\psi_i\in\mathbb{R}^2$ by the formula:

$$\mathcal{H}^{(\kappa)}(x_h,\tau_h+\tau\,\psi_i) \quad = \quad \mathcal{H}^{(\kappa)}(x_h,\tau_h)\;+\;\mathcal{H}_i^{(\kappa)}(x_h,\tau_h+\tau\,\psi_i)\;-\;\mathcal{H}_i^{(\kappa)}(x_h,\tau_h) \quad .$$

Consequently, a full sweep of the mesh is necessary but once to provide an initial value for the objective function $\mathcal{H}^{(\kappa)}$. Each line search can then be performed by calculating only the offsets $\mathcal{H}_i^{(\kappa)}$. If the dual parameter τ_h is sought in some Raviart-Thomas space, the procedure can stay basically unchanged: only the patches Ω_i differ.

One step of the optimisation procedure consists in a successive relaxation of all the degrees of freedom, which define the dual parameter. The outer iteration may be terminated as soon as a significant improvement of the error estimate $\mathcal{H}^{(\kappa)}(x_h,\cdot)$ can no longer be achieved. In our numerical experiments we have simply performed a fixed number of relaxation sweeps without monitoring the progress of the optimisation. The number of sweeps is indicated by a superscript to

the name of the dual variable. The notation is essentially the same as in the previous section: $\hat{\sigma}_1^{(n)}$ for instance denotes the result of n optimisation loops applied to the vector field $\hat{\sigma}_1 \in R_c^1(\mathcal{M}_h)$, the solution of the variational statement (5.1), while $\tau^{(n)} \in (P_c^1(\mathcal{M}_h))^2$ designates the outcome of n optimisation steps applied to the smoothed gradient $\Pi(\nabla x_h)$ of the primal approximation.

Figure 5.4: The impact of Optimisation on the Efficiency Indices I_{eff}

Estimate	Example	Level 0	Level 1	Level 2	Level 3	Level 4
$\mathcal{R}_h(x_h)$	A	4.24	4.39	4.44	4.47	4.50
$\mathcal{H}^*(x_h, \tau_h^{(10)})$	A	7.20	10.34	14.00	20.18	28.12
$\mathcal{H}^*(x_h, \rho_0^{(10)})$	A	7.24	10.28	13.89	19.84	27.24
$\mathcal{H}^*(x_h, \hat{\rho}_0^{(10)})$	A	7.29	10.32	13.91	19.86	27.26
$\mathcal{H}^*(x_h, \sigma_1^{(10)})$	A	6.95	9.99	13.59	19.54	26.94
$\mathcal{H}^*(x_h, \hat{\sigma}_1^{(10)})$	A	7.10	10.27	13.81	19.67	27.01
$\mathcal{H}^*(x_h, \rho_1^{(10)})$	A	4.66	6.52	8.63	12.23	16.71
$\mathcal{H}^*(x_h, \hat{\rho}_1^{(10)})$	A	5.06	7.08	9.67	13.89	20.14
$\mathcal{R}_h(x_h)$	B	4.49	4.52	4.56	4.59	4.59
$\mathcal{H}^*(x_h, \tau_h^{(10)})$	B	2.82	2.87	2.96	3.06	3.20
$\mathcal{H}^*(x_h, \rho_0^{(10)})$	B	3.00	2.98	2.97	2.97	2.97
$\mathcal{H}^*(x_h, \hat{\rho}_0^{(10)})$	B	3.04	3.02	3.02	3.10	3.20
$\mathcal{H}^*(x_h, \sigma_1^{(10)})$	B	2.75	2.73	2.73	2.73	2.73
$\mathcal{H}^*(x_h, \hat{\sigma}_1^{(10)})$	B	2.87	2.99	2.93	2.91	2.91
$\mathcal{H}^*(x_h, \rho_1^{(10)})$	B	1.22	1.15	1.10	1.07	1.05
$\mathcal{H}^*(x_h, \hat{\rho}_1^{(10)})$	B	2.09	3.00	4.49	6.71	11.07
$\mathcal{R}_h(x_h)$	C	4.16	4.37	4.45	4.48	4.49
$\mathcal{H}^*(x_h, \tau_h^{(10)})$	C	6.42	7.30	7.53	7.63	7.68
$\mathcal{H}^*(x_h, \rho_0^{(10)})$	C	6.43	7.12	7.18	7.19	7.19
$\mathcal{H}^*(x_h, \hat{\rho}_0^{(10)})$	C	6.47	7.15	7.22	7.23	7.25
$\mathcal{H}^*(x_h, \sigma_1^{(10)})$	C	6.16	6.84	6.90	6.91	6.91
$\mathcal{H}^*(x_h, \hat{\sigma}_1^{(10)})$	C	6.33	7.09	7.07	7.02	6.98
$\mathcal{H}^*(x_h, \rho_1^{(10)})$	C	2.75	1.89	1.43	1.22	1.11
$\mathcal{H}^*(x_h, \hat{\rho}_1^{(10)})$	C	3.64	3.92	4.57	6.44	11.23

Comparing the entries in the tables 5.3 and 5.4 we note, that the efficiency indices of those hypercycle estimates, which are based on the solution of the dual formulation, are hardly affected by a subsequent optimisation of the error bound. Those estimates, however, which have been computed with the help of a projection respectively a smoothing of the gradient ∇x_h, benefit significantly from the post-processing. On coarser meshes the resulting error bounds are even comparable to those results obtained from approximating the dual solution. On finer meshes a fixed number of optimisation steps is insufficient to guarantee uniform bounds on the efficiency indices, let alone asymptotically exact estimates. In particular, the lack of convergence becomes apparent, when Raviart-Thomas elements of higher order are employed. As we should be able to obtain superior results, if we continued the optimisation procedure but long enough, it might be worthwhile to look for improved optimisation algorithms. Our relaxation method has been implemented with a view to simplicity and universal applicability: it has no other merit.

If we combine it with several optimisation steps, the gradient recovery scheme defined by (5.1) yields results, that are only slightly inferior to those hypercycle estimates based on lowest order elements of Raviart-Thomas type. Considering the fact, that at least the trial space $B_c^1(\mathcal{M}_h)$ features much more degrees of freedom than the ansatz $(P_c^1(\mathcal{M}_h))^2$, the quality of the error bounds $\mathcal{H}^*(x_h, \tau_h^{(10)})$ is quite remarkable. We may surmise, that our regularity assumptions on

the data of the Laplace problem, which cause the analytical solution of the dual formulation to be contained in the space $H^1(\Omega, \mathbb{R}^2)$, can be held responsible for the lacking edge of discontinuous trial fields.

5.2.5 Alternative Approaches

The numerical results we have compiled in the tables C.5, C.6 and C.7 indicate, that the residual $M_R(x_h, \cdot)$ in the second of the two duality relations (1.12) is not properly accounted for, when the dual parameter is computed by smoothing the gradient ∇x_h of the numerical solution with the help of the projection (5.1). The effect is particularly pronounced when higher order elements are employed to represent the dual parameter. We may try and reduce the incriminated residual by modifying the projection in accordance with the following rationale: The analytical solution $\sigma \in H_{\mathrm{div}}(\Omega)$ of the dual formulation meets the condition

$$(\nabla\cdot \tau, \nabla\cdot \sigma)_\Omega \quad = \quad -(f, \nabla\cdot \tau)_\Omega \qquad ; \qquad \tau \in H_{\mathrm{div}}(\Omega) \ .$$

Hence, it seems reasonable to combine the above equation with (5.1) and replace the L^2-projection with some sort of H_{div}-projection. To warrant an uniform bound on the numerical complexity of the smoothing algorithm, we must ensure, however, that the condition number of the projection matrix stays bounded. One possibility consists in scaling a part of the matrix with the square of the mesh parameter h to suppress the impact of the first derivatives:

$$\langle \tau_h, \tilde\sigma_h \rangle_\Omega \ + \ \alpha_0 h^2 (\nabla\cdot \tau_h, \nabla\cdot \tilde\sigma_h)_\Omega \quad = \quad -(\alpha_0 h^2 f \ + \ x_h, \nabla\cdot \tau_h)_\Omega \quad ; \quad \tau_h \in Y_h \ .$$

Hereby, $Y_h \subset H_{\mathrm{div}}(\Omega)$ denotes a suitable trial space. In our experiments the parameter $\alpha_0 > 0$ was defined by the requirement $\alpha_0 h^2 = 0.02$ to be met on the coarsest level of the triangulation. The notation, we use in the tables C.11 through C.13 and in the summary, figure 5.5, is basically the same as in the other tables: A tilde indicates, that the vector field has been obtained by the H_{div}-projection we have just specified. Fields without a tilde have been obtained by solving the dual formulation numerically. As in section 5.2.3 the greek letters and the subscripts signify our choice of the ansatz respectively the polynomial order of the trial fields.

The hypercycle estimate (2.8) has been introduced with a view to avoiding the dual norm which is present in the more accurate error bound (2.7). Generally speaking, we are unable to assess said norm, whence the latter estimate is not practicable. However, we may control the dual norm by the same device we employ in the computation of conventional error estimates, which are based on element residuals. Assuming the field $\sigma_h \in H_{\mathrm{div}}(\Omega)$ satisfies the admissibility constraint $\nabla\cdot \sigma_h + P_k f = 0$ *exactly*, we can exploit the best approximation property of the L^2-projection:

$$| \nabla\cdot \sigma_h + f |_{\Omega,-1} \quad = \quad \sup_{v \in H_0^1(\Omega)} \frac{(f - P_k f, v - P_0 v)_\Omega}{\| \nabla v \|_\Omega} \quad \leq \quad \Big\{ \sum_{M \in \mathcal{M}_h} C_M^2 \, h_M^2 \, \| f - P_k f \|_M^2 \Big\}^{1/2} \ .$$

A priori, the values of the local interpolation constants $C_M > 0$ are unknown. While the choice of safe upper bounds for these quantities can seriously deteriorate the accuracy of a residual based error estimator, the generalised hypercycle estimate (2.7) is not affected by too pessimistic estimates: the above bound on the dual norm is but a higher order perturbation, so the choice of the interpolation constants becomes inconsequential on finer meshes. Let us define:

$$M_{-1}^{(\kappa)}(x_h, \tau) \quad := \quad \big(1 + \kappa \big) \Big\{ \sum_{M \in \mathcal{M}_h} h_M^2 \, \| f + \nabla\cdot \tau \|_M^2 \Big\} \tag{5.3}$$

for any vector field $\tau \in H_{\mathrm{div}}(\Omega)$. With a view to [49] we can assert, that all the interpolation constants C_M are smaller than 1. Hence, the above quantity constitutes an upper bound on the second part of the hypercycle estimate (2.7) - if only under the condition, that the field τ meet the requirement $\nabla\cdot \tau + P_0 f = 0$. The most straightforward method of obtaining such a field consists in solving the dual formulation

$$J^*(\tau) \ = \ F^*(\tau) \ + \ G^*(\nabla\cdot \tau) \quad \longrightarrow \quad \min$$

Figure 5.5: Alternative Methods of Computing Hypercycle Estimates

Estimate	Example	Level 0	Level 1	Level 2	Level 3	Level 4
$\mathcal{H}^*_{-1}(x_h,\rho_0)$	A	1.63	1.56	1.48	1.44	1.40
$\mathcal{H}^*_{-1}(x_h,\sigma_1)$	A	1.33	1.26	1.18	1.14	1.10
$\mathcal{H}^*_{-1}(x_h,\sigma_1)$	A	1.20	1.16	1.11	1.08	1.06
$\mathcal{H}^*(x_h,\tilde\rho_0)$	A	7.25	10.30	19.42	19.89	27.29
$\mathcal{H}^*(x_h,\tilde\rho_0^{(10)})$	A	7.24	10.28	13.89	19.85	27.23
$\mathcal{H}^*(x_h,\tilde\sigma_1)$	A	6.95	9.99	13.60	19.57	26.95
$\mathcal{H}^*(x_h,\tilde\sigma_1^{(10)})$	A	6.95	9.98	13.59	19.55	26.92
$\mathcal{H}^*(x_h,\tilde\rho_1)$	A	4.67	6.54	8.69	12.37	17.02
$\mathcal{H}^*(x_h,\tilde\rho_1^{(10)})$	A	4.66	6.52	8.64	12.26	16.75
$\mathcal{H}^*_{-1}(x_h,\rho_0)$	B	1.35	1.30	1.27	1.27	1.25
$\mathcal{H}^*_{-1}(x_h,\sigma_1)$	B	1.11	1.05	1.03	1.01	1.01
$\mathcal{H}^*_{-1}(x_h,\rho_1)$	B	1.01	1.00	1.00	1.00	1.00
$\mathcal{H}^*(x_h,\tilde\rho_0)$	B	3.02	3.05	3.14	3.30	3.56
$\mathcal{H}^*(x_h,\tilde\rho_0^{(10)})$	B	3.00	2.98	2.98	3.01	3.05
$\mathcal{H}^*(x_h,\tilde\sigma_1)$	B	2.76	2.78	2.84	2.95	3.12
$\mathcal{H}^*(x_h,\tilde\sigma_1^{(10)})$	B	2.75	2.73	2.73	2.74	2.77
$\mathcal{H}^*(x_h,\tilde\rho_1)$	B	1.36	1.54	1.91	2.57	3.83
$\mathcal{H}^*(x_h,\tilde\rho_1^{(10)})$	B	1.28	1.28	1.43	1.64	1.85
$\mathcal{H}^*_{-1}(x_h,\rho_0)$	C	1.55	1.44	1.36	1.32	1.30
$\mathcal{H}^*_{-1}(x_h,\sigma_1)$	C	1.28	1.16	1.08	1.04	1.02
$\mathcal{H}^*_{-1}(x_h,\rho_1)$	C	1.09	1.02	1.01	1.00	1.00
$\mathcal{H}^*(x_h,\tilde\rho_0)$	C	6.43	7.15	7.24	7.28	7.33
$\mathcal{H}^*(x_h,\tilde\rho_0^{(10)})$	C	6.43	7.12	7.19	7.20	7.21
$\mathcal{H}^*(x_h,\tilde\sigma_1)$	C	6.16	6.85	6.93	6.96	7.00
$\mathcal{H}^*(x_h,\tilde\sigma_1^{(10)})$	C	6.16	6.84	6.90	6.91	6.92
$\mathcal{H}^*(x_h,\tilde\rho_1)$	C	2.78	2.12	2.12	2.61	3.81
$\mathcal{H}^*(x_h,\tilde\rho_1^{(10)})$	C	2.76	1.99	1.76	1.75	1.81

of the Dirichlet problem with an ansatz of Raviart-Thomas type. Once the solution of the dual mixed discretisation has been computed, further processing of the hypercycle estimate

$$\mathcal{H}^{(\kappa)}_{-1}(x_h,\tau) \quad := \quad M^{(\kappa)}_D(x_h,\tau) \;+\; M^{(\kappa)}_{-1}(x_h,\tau) \tag{5.4}$$

is very difficult, however: any optimisation algorithm designed to improve the hypercycle estimate $\mathcal{H}^{(\kappa)}_{-1}$ is required to observe the admissibility constraint $\nabla\cdot\tau + P_0 f = 0$, which introduces a global coupling between all the degrees of freedom associated with the dual parameter τ. In consequence, fast line search procedures as the one described in section 5.2.4 can no longer be employed. The optimal equilibration parameter κ^* is determined by the formula (5.2).

5.3 Energy Error Estimates for the Obstacle Problem

The analysis of the alternative hypercycle estimate originally introduced in the technical report [41] has proved difficult and the results, we have obtained in section 3.3.2, remain somewhat unsatisfactory. Our efforts have been hampered by the fact, that the analytical solution is but of limited smoothness, even if the data of the obstacle problem is analytic. With a view to the

deliberations, we have offered in paragraph 2.2.8, we may expect the alternative error bound (2.19) to perform better than the generalised hypercycle estimate (2.16). However, we have been unable to quantify the discrepancies between both error estimators. In the following paragraphs we shall try and indicate the differences by a number of very simple numerical experiments, which involve the same meshes we have already employed in section 5.2. We will moreover indicate a procedure, by which an upper bound for the more precise hypercycle estimate (2.15) can be obtained from the solution of a suitably modified dual problem.

5.3.1 Obtaining the Analytical Solution

Obtaining the analytical solution of an obstacle problem is practically impossible, unless the data of the problem admits a trivial solution or the contact set is actually void. However, in certain cases, essentially one-dimensional in nature, we may construct the analytical solution in such a way, that the function depends but on a single parameter, which we can determine with a simple fixed point iteration. Let us suppose, that the domain $\Omega \subset \mathbb{R}^n$ is a circle with radius 1 centered around the origin. Let us further suppose, that the obstacle be flat and that the forcing function be constant $f \equiv 4$. If the obstacle function $\psi \equiv \psi_0$ were smaller than -1, the solution $u \in H_0^1(\Omega)$ of the variational problem (2.12) would be a simple parabola. Let us therefore assume $\psi_0 \in (-1, 0)$. Employing radial coordinates we may stipulate an ansatz of the form:

$$u(r) \quad := \quad \begin{cases} \psi_0 & ; \quad 0 \leq r \leq r_0 \\ 1 - r^2 + \zeta_0 \ln(r) & ; \quad \text{else} \end{cases} \quad .$$

We note, the above function meets the homogeneous Dirichlet boundary conditions. Moreover, the Laplace equation for radially symmetric functions is met on an annulus defined by r_0:

$$-\frac{1}{r}\frac{\partial}{\partial r}\left(r \frac{\partial}{\partial r} u \right) \quad = \quad f \quad ; \quad r \in (r_0, 1) \quad .$$

In accordance with the regularity theory for variational inequalities we can expect our solution to be Hölder continuous: $u \in C^{1,1}([0,1])$. Hence, we must ensure that both our ansatz and it's first derivative are continuous at the boundary of the coincidence set as specified by the a priori unknown parameter r_0. We conclude:

$$\psi_0 \quad = \quad 1 - r_0^2 + \zeta_0 \ln(r_0) \qquad \wedge \qquad 0 \quad = \quad -2r_0 + \zeta_0 r_0^{-1} \quad .$$

Eliminating the parameter ζ_0 we find a nonlinear equation for the radius of the coincidence set:

$$1 - \psi_0 \quad = \quad r_0^2 - r_0^2 \ln(r_0^2) \quad .$$

It's solution can be easily obtained up to machine-precision with the help of the following iterative scheme, which we may start e. g. with the initial choice $\rho^{(0)} = 10$ and terminate, once the distance between two succeeding iterates has become smaller than some prescribed threshold:

$$\rho^{(n+1)} \quad := \quad \Phi(\rho^{(n)}) \quad := \quad \frac{1 + \ln(\rho^{(n)})}{1 - \psi_0} \quad .$$

We note, that $0 < \Phi'(\rho) < 1$ holds for any argument $\rho > 1$. Moreover, $\Phi(1)$ is larger than 1. In consequence, the above iteration schemes admits an unique fixed point $\rho^* > 1$. Once this point has been found, we can recover the parameters r_0 and ζ_0 by:

$$r_0 \quad = \quad \frac{1}{\sqrt{\rho^*}} \qquad ; \qquad \zeta_0 \quad = \quad \frac{2}{\rho^*} \quad .$$

5.3.2 Residual based Error Estimates

The conventional approach to estimating the residual in some dual norm cannot be applied to the obstacle problem for basically two reasons: the very residual does not provide any sensible information on the accuracy of our numerical solution and secondly we are unable to assess the norm, since that function lacks the best approximation property. To employ a conventional direct

error estimator nevertheless, it is necessary to replace the constrained minimisation task with an unconstrained penalty formulation of the obstacle problem. Let us assume, that $\epsilon : \Omega \longrightarrow \mathbb{R}^+$ is a "small" positive function and that $u_\epsilon \in H_0^1(\Omega)$ solves the following nonlinear problem:

$$\int_\Omega \Big\{ \, (\nabla v)^T \, \nabla x_\epsilon \, + \, \frac{v}{\epsilon} \min\{0, x_\epsilon - \psi\} \, \Big\} \, d\xi \quad = \quad 0 \qquad ; \qquad v \in H_0^1(\Omega) \ .$$

As with all direct a posteriori error estimators we must suppose, that the numerical approximation $x_h \in H_0^1(\Omega) \cap P_c^1(\mathcal{M}_h)$ solves the finite dimensional counterpart of the above statement *exactly:*

$$\langle \, \nabla x_h, \nabla v_h \, \rangle_\Omega \, + \, (\, \epsilon^{-1} \min\{0, x_h - \psi\}, v_h \,)_\Omega \; = \; (\, f, v_h \,)_\Omega \tag{5.5}$$

with $v_h \in H_0^1(\Omega) \cap P_c^1(\mathcal{M}_h)$ denoting an arbitrary test function. Exploiting the so called *Galerkin orthogonality* of the numerical solution x_h described by (5.5) along with the usual approximation results for the trial space $P_c^1(\mathcal{M}_h)$, we can indeed find a computable upper bound on the energy error in terms of a nonlinear residual. Let $g^- := \min\{0, g\} \in H^1(\Omega)$ denote the negative part of the function $g \in H^1(\Omega)$. (The regularity of g^- is proved e. g. in [101]). We contend:

$$\begin{aligned}
| \, x_\epsilon - x_h \, |_{\Omega,1}^2 \quad &= \quad (\, f, x_\epsilon - x_h \,)_\Omega - \langle \, \nabla x_h, \nabla (x_\epsilon - x_h) \, \rangle_\Omega - \, (\, \epsilon^{-1} (x_h - \psi)^-, \, x_\epsilon - x_h \,)_\Omega \\
&- \quad \big(\, \epsilon^{-1} (x_h - \psi)^- - \epsilon^{-1} (x_\epsilon - \psi)^-, \, (x_h - \psi) - (x_\epsilon - \psi) \, \big)_\Omega \\
&\leq \quad \sum_{M \in \mathcal{M}_h} \int_M \big(\, f - \epsilon^{-1} (x_h - \psi)^- \, \big) \big(\, x_\epsilon - x_h \, \big) \, d\xi \, + \, \oint_{\partial M} \frac{\partial x_h}{\partial n} \big(\, x_\epsilon - x_h \, \big) \, ds \\
&\leq \quad \sum_{M \in \mathcal{M}_h} C_M \Big\{ \, h_{_M} \, \Big\| \, f - \frac{(x_h - \psi)^-}{\epsilon} \, \Big\|_{_M} + \sqrt{h_{_M}} \, \Big\| \Big[\frac{\partial x_h}{\partial n} \Big] \Big\|_{\partial M} \Big\} \, | \, x_\epsilon - x_h \, |_{_{M,1}} \ .
\end{aligned}$$

Hence, as far as the penalty formulation is concerned, the final a posteriori error estimate reads:

$$\mathcal{R}_{h,\epsilon}(x_h) \quad := \quad \Big\{ \sum_{M \in \mathcal{M}_h} C_M^2 \Big\{ \, h_{_M}^2 \, \Big\| \, f - \frac{(x_h - \psi)^-}{\epsilon} \, \Big\|_{_M}^2 + \, h_{_M} \, \Big\| \Big[\frac{\partial x_h}{\partial n} \Big] \Big\|_{\partial M}^2 \Big\} \Big\}^{1/2} \ .$$

However, there is also a consistency error to be taken into account. With a view to the necessary optimality condition for the solution $x_0 \in V_\psi$ of the obstacle problem (2.12) we may state:

$$\begin{aligned}
| \, x_\epsilon - x_0 \, |_{\Omega,1}^2 \quad &= \quad (\, f, x_\epsilon - x_0 \,)_\Omega - \langle \, \nabla x_0, \nabla (x_\epsilon - x_0) \, \rangle_\Omega - \, (\, \epsilon^{-1} (x_\epsilon - \psi)^-, \, x_\epsilon - x_0 \,)_\Omega \\
&\leq \quad (\, f, (x_\epsilon - \psi)^- \,)_\Omega - \langle \, \nabla x_0, \nabla (x_\epsilon - \psi)^- \, \rangle_\Omega - \, (\, \epsilon^{-1} (x_\epsilon - \psi)^-, \, x_\epsilon - x_0 \,)_\Omega \\
&= \quad \langle \, \nabla (x_\epsilon - x_0), \nabla (x_\epsilon - \psi)^- \, \rangle_\Omega \, + \, (\, \epsilon^{-1} (x_\epsilon - \psi)^-, \, x_0 - \psi \,)_\Omega \\
&\leq \quad | \, x_\epsilon - x_0 \, |_{\Omega,1} \, | \, (x_\epsilon - \psi)^- \, |_{\Omega,1} \quad .
\end{aligned} \tag{5.6}$$

In the course of the above computations we have used the fact twice, that we can decompose the function x_ϵ into two contributions, one of which is an admissible test function for the obstacle problem: $x_\epsilon = (x_\epsilon - \psi)^- + (\psi + (x_\epsilon - \psi)^+)$. We have used furthermore, that $(x_\epsilon - \psi)^-$ and $(x_\epsilon - \psi)^+$ are mutually orthogonal. Due to the latter fact we find:

$$| \, (x_\epsilon - \psi)^- \, |_{\Omega,1}^2 \; + \; \| \, \epsilon^{-1/2} \, (x_\epsilon - \psi)^- \, \|_\Omega^2 \quad = \quad (\, f + \Delta\psi, \, (x_\epsilon - \psi)^- \,)_\Omega \quad ,$$

for $(x_\epsilon - \psi)^- \in H_0^1(\Omega)$ is an admissible test function and $\psi \in H^2(\Omega)$ holds by assumption. The above result implies both a bound on the norm an on the seminorm in terms of the data:

$$\| \, \epsilon^{-1/2} \, (x_\epsilon - \psi)^- \, \|_\Omega^2 \quad \leq \quad \| \, \epsilon^{1/2} \, (f + \Delta\psi) \, \|_\Omega \, \| \, \epsilon^{-1/2} \, (x_\epsilon - \psi)^- \, \|_\Omega \quad \leq \quad \| \, \epsilon^{1/2} \, (f + \Delta\psi) \, \|_\Omega^2 \ .$$

Combining this latter estimate with (5.6) we conclude:

$$| \, x_\epsilon - x_0 \, |_{\Omega,1} \quad \leq \quad \mathcal{S}_\epsilon(f, \psi) \quad := \quad \| \, \epsilon^{1/2} \, (f + \Delta\psi) \, \|_\Omega \quad . \tag{5.7}$$

With a view to (5.7) we must ensure, that the function ϵ is coupled to the local mesh width in accordance to $\epsilon \simeq h^2$ in order to ensure, that the consistency error has the asymptotic behaviour under mesh refinement as the error bound $\mathcal{R}_{h,\epsilon}$ for the penalty formulation (5.5). The final a posteriori error estimate for the obstacle problem consists of both contributions:

$$| x_h - x_0 |_{\Omega,1} \quad \leq \quad \mathcal{S}_\epsilon(f,\psi) \quad + \quad \mathcal{R}_{h,\epsilon}(x_h) \quad . \tag{5.8}$$

5.3.3 Miscellaneous Remarks on the Experiments

Both the generalised hypercycle estimate for the Laplace and for the obstacle problem depend on the equilibration parameter κ in a very simple fashion: once the dual variable τ has been determined, the optimal value κ^* can be obtained a posteriori with the help of formula (5.2). Unfortunately, the dependency of the alternative error estimate (2.19) on the parameter κ is much more involved, as we have learned in the sections 2.2.7 and 2.2.8. The sharpest error bound can certainly be obtained by minimising the hypercycle estimate

$$\begin{aligned} \mathcal{H}^{(\kappa)}(x_h,\tau) \quad := \quad & \frac{1+\kappa}{\kappa} \, \| \nabla x - \tau \|_\Omega^2 \quad + \quad \frac{1+\kappa}{\lambda_0^2} \, \| \nabla\cdot \tau + f \|_\Omega^2 \\ & - \quad \frac{1+\kappa}{\lambda_0^2} \, \left\| 0 \wedge \left(\lambda_0^2 \, \frac{x-\psi}{1+\kappa} \, + \, \nabla\cdot \tau \, + \, f \right) \right\|_\Omega^2 \end{aligned}$$

with respect to both the vector field τ and the parameter κ. From a practical point of view a simultaneous minimisation algorithm is difficult to implement, however. A more conventional minimisation algorithm similar to the one described in section 5.2.4 will allow chiefly for local couplings between the degrees of freedom associated with the field τ, while changes in κ will affect all these quantities in an equal measure. Adjusting the equilibration parameter κ while sweeping the mesh may therefore severely obstruct the minimisation process. An alternating procedure, which combines relaxation sweeps to improve τ with a fixed point iteration to determine κ^* may possibly be the best solution.

Figure 5.6: Finding the Optimal Equilibration Parameter

Supplied: $\kappa > 0$, $\varepsilon > 0$, $\tau_h^* \in H_{\mathrm{div}}(\Omega)$, $x_h \in V_\psi$

Loop: {

1) **Set:** $\kappa_{\mathtt{old}} := \kappa$.

2) **Compute the hypercycle estimate:**

$$\begin{aligned} M_R^{(\kappa)}(x_h,\tau_h^*) \quad := \quad & \frac{1+\kappa}{\lambda_0^2} \, \| \nabla\cdot \tau_h^* + f \|_\Omega^2 \\ & - \quad \frac{1+\kappa}{\lambda_0^2} \, \left\| 0 \wedge \left(\lambda_0^2 \, \frac{x-\psi}{1+\kappa} \, + \, \nabla\cdot \tau_h^* \, + \, f \right) \right\|_\Omega^2 \end{aligned}$$

$$M_D^{(\kappa)}(x_h,\tau_h^*) \quad := \quad \frac{1+\kappa}{\kappa} \, \| \nabla x_h - \tau_h^* \|_\Omega^2 \ .$$

3) **Determine the new equilibration parameter:**

$$\kappa \quad := \quad \sqrt{\frac{\kappa \, M_D^{(\kappa)}(x_h,\tau_h^*)}{M_R^{(\kappa)}(x_h,\tau_h^*)}} \quad .$$

4) **If** $|\kappa - \kappa_{\mathtt{old}}| < \varepsilon$ **holds, terminate the iteration.**

}

In all the experiments, which involve the alternative error estimate introduced in section 2.2.5, we have employed a far simpler procedure to find an adequate equilibration parameter κ^*: We compute a vector field $\tau_h \in H_{\mathrm{div}}(\Omega)$ either by solving the dual formulation of the obstacle problem (see section 4.3.2) or by projecting the gradient of the primal approximation into a suitable trial space. After some optional post-processing the resulting vector field τ_h^* is frozen. If the initial vector field is modified by a minimisation procedure, the algorithm is executed with κ being kept fixed. Once the vector field τ_h^* has been determined, the iteration scheme depicted in figure 5.6 is used to compute a suitable equilibration parameter.

Exploiting the Galerkin Orthogonality

The hypercycle estimates for both the Laplace (2.7) and the obstacle problem (2.15) are not computable per se, for they involve dual norms of certain residual expressions. By invoking the inequality (1.4) of Poincaré-Friederich type we can turn them into computable error bounds in accordance with proposition 1.2. With respect to the Laplace problem an alternative approach has been outlined in section 5.2.5. If we assume, that the vector field $\tau_h^* \in H_{\mathrm{div}}(\Omega)$ solves the dual formulation *exactly*, the second part of the estimate (2.7) can be controlled thanks to the best approximation property of the divergence: $\nabla\cdot\,\tau_h^* + P_0 f = 0$.

A similar strategy may be applied to the obstacle problem. To find a computable bound for the second part of the hypercycle estimate (2.15) we must ensure, the dual parameter τ_h^* meets the condition $\nabla\cdot\,\tau_h^* + P_0 f = 0$ on some domain $\Omega_h \subset \Omega$ with the property: $\Omega_x \supset \Omega \setminus \Omega_h$. To obtain a sharp hypercycle estimate, the domain Ω_h should be possibly small. We define:

$$\mathcal{M}_h^0 \;:=\; \left\{\, M \in \mathcal{M}_h \;\middle|\; M \cap \Omega_x \neq M \,\right\} \quad .$$

The set $\mathcal{M}_h^0$ contains all but those elements, which are completely contained within the coincidence set Ω_x of the primal approximation x_h. (Regarding the definition of Ω_x we refer to section 2.2.3.) We introduce a new obstacle by reducing the support of the obstacle function ψ:

$$E \;:=\; cl\Big(\; \Omega \;\setminus \bigcup_{M\in\mathcal{M}_h^0} cl(M) \;\Big) \quad .$$

Since E is a simplicial domain, the admissibility constraint $\nabla\cdot\,\sigma_h + P_0 f = 0$, so to speak, can be satisfied exactly on its complement, if we but employ a trial space of Raviart-Thomas type to compute the numerical solution σ_h of the following variational problem:

$$\frac{1}{2}\,\|\,\sigma_h\,\|^2 \;\longrightarrow\; \min \qquad \text{s.t.:} \qquad \nabla\cdot\,\sigma_h \,+\, f\,\Big|_E \;\leq\; 0 \quad .$$

The resulting hypercycle estimate is basically the same as (5.4). However, with a view to the original error bound (2.15) the residual contribution (5.3) must be replaced with:

$$M_{-1}^{(\kappa)}(x_h,\tau) \;:=\; \big(1+\kappa\big)\,\Big\{\, \sum_{M\in\mathcal{M}_h^0} h_M^2\,\|\,f+\nabla\cdot\,\tau\,\|_M^2 \,\Big\} \qquad ; \qquad \tau \in H_{\mathrm{div}}(\Omega) \quad .$$

The optimal equilibration parameter κ^* can be determined a posteriori by the formula (5.2).

Solving the Penalty Formulation

If we wish to employ a residual based error estimator for the obstacle problem, we must replace the constrained minimisation problem with a suitable penalty formulation. One such variational statement we have analysed in section 5.3.2. Since the problem (5.5) is nonlinear, we cannot use a conventional multilevel method to compute its solution $x_h \in P_c^1(\mathcal{M}_h)$. One possibility consists in operating the multilevel algorithm in the so called FAS mode [28]. Another possibility consists in implementing a fixed point iteration with the help of an appropriate preconditioner for the stiffness matrix. We have seen in section 5.3.2, that the penalty function $\epsilon : \Omega \longrightarrow \mathbb{R}^+$ must be coupled to the mesh width in order to control the consistency error. To ensure the error estimate is not dominated by the consistency error $\mathcal{S}_\epsilon(f,\psi)$ this function should be smaller than $\epsilon_0\, h_M^2$ on any element $M \in \mathcal{M}_h$ within the top layer of the mesh.

Our attempts of using a FAS multilevel scheme on unstructured meshes have failed repeatedly, when the penalty function ϵ has been computed be averaging the local mesh parameters h_M on the finest mesh. If the mesh had been obtained through successive global mesh refinements and ϵ was calculated on the coarsest mesh, we have been able to run the multilevel iteration successfully. However, in that latter case the scaling parameter $\epsilon_0 > 0$ must account for the mesh refinement. Since decreasing ϵ_0 has led to severe performance degradations, we have settled on the alternative approach to solving the penalty formulation (5.5).

Let $m \in \mathbb{N}$ designate the degrees of freedom associated with the trial space $P_c^1(\mathcal{M}_h) \cap H_0^1(\Omega)$. The shape functions of this space shall be denoted by $\{\phi_i\}_{i=1}^m$. The degrees of freedom can be identified with the set of those vertices $\{x_i\}_{i=1}^m$, which constitute the interior to the mesh $\mathcal{M}_h$. The enumeration of these points corresponds to a local enumeration scheme on an element by element basis, which we will refer to by a double index. In the same way, we shall describe the collocation points $\{\hat{x}_{M,l}\}_{l=1}^L \subset cl(M)$ and the associated weights $\{\Theta_{M,l}\}_{l=1}^L$ of the numerical cubature scheme, we employ on some element $M \in \mathcal{M}_h$. Let us specify a penalty function $\epsilon \in P_c^1(\mathcal{M}_h)$ by:

$$\epsilon(x_i) \quad := \quad \sum_{\partial M \ni x_i} \epsilon_0 \, h_M^2 \qquad ; \qquad i \in \{1, \dots m\} \quad .$$

In all of our experiments we have used the value $\epsilon_0 = 0.2$. With a view to the variational problem (5.5) we define the following nonlinear mapping $f : \mathbb{R}^m \longrightarrow \mathbb{R}^m$ component by component:

$$f_k(\xi) \quad := \quad \sum_{M \in \mathcal{M}_h} \sum_{l=1}^{L} \Theta_{M,l} \, \phi_k(\hat{x}_{M,l}) \, \Big\{ \; f(\hat{x}_{M,j}) \; - \; \Big\{ \sum_{j=1}^{3} \epsilon(x_{M,j}) \, \phi_{M,j}(\hat{x}_{M,l}) \Big\}^{-1} \; \times$$

$$\max \Big\{ \, 0 \, , \, \sum_{j=1}^{3} (\xi_{M,j} - \psi(x_{M,j})) \, \phi_{M,j}(\hat{x}_{M,l}) \Big\} \; \Big\} \qquad ; \qquad k \in \{1, \dots m\} \quad .$$

The stiffness matrix shall be designated $S \in \mathbb{R}^{m \times m}$. We have implemented the following iterative scheme to compute the solution of the problem (5.5):

$$\xi^{(n+1)} \quad := \quad (\, 1 \, - \, \omega_h \,) \, \xi^{(n)} \; + \; \omega_h \, S^{-1} \, f(\xi^{(n)}) \qquad ; \qquad n \in \mathbb{N}_0$$

starting the iteration with the vector $\xi^{(0)} = 0$ and using a mesh dependent damping parameter $\omega_h \in [5.2\text{e-}4, 7\text{e-}2]$. Despite the fact, that the action of the matrix S^{-1} is computed up to machine precision with the help of a multilevel solver, the performance of the above fixed point scheme has been disappointing. Below, the map $M_S : \mathbb{R}^m \times \mathbb{R}^m \longrightarrow \mathbb{R}^m$ may designate the action $M_S(\xi, f)$ of just one multilevel V-cycle designed for computing $S^{-1} f$ on the initial vector ξ. We define:

$$\xi^{(n+1)} \quad := \quad (\, 1 \, - \, \omega_h \,) \, \xi^{(n)} \; + \; \omega_h \, M_S(\, \xi^{(n)}, \, f(\xi^{(n)}) \,) \qquad ; \qquad n \in \mathbb{N}_0$$

again starting the iteration with $\xi^{(0)} = 0$. Using mesh dependent damping parameters ω_h this latter fixed point scheme has been superior in terms of computational effort both to the former scheme and the FAS multilevel solver. Unfortunately, even by aggressive tuning of the relaxation parameters ω_h we fail to make the method competitive with a conventional multilevel scheme (see e. g. [106]) for the constrained variational formulation. In figure 5.7 we give an indication of the numerical complexity.

Figure 5.7: Solving the Obstacle Problem by Penalisation

	Level 0	Level 1	Level 2	Level 3	Level 4
m	433	1849	7633	31009	124993
ω_h	0.070	0.033	0.0085	0.0022	0.00052
# Iterations	209	485	1704	5927	22144

5.3.4 A few Remarks on the Numerical Results

In the figure 5.8 we summarise our findings, as far as the computable hypercycle estimates (2.16) and (2.19) are concerned. We have employed three different trial spaces to produce the dual parameter, which we have determined either by a projection of the gradient ∇x_h into the space $H_{\mathrm{div}}(\Omega)$ or by solving the dual formulation of the obstacle problem. Moreover, the line search procedure described in section 5.2.4 has been adapted to the obstacle problem in order to improve the error bounds by post-processing the dual parameter. The notation we use in figure 5.8 follows the very conventions, we have established in the previous sections.

Figure 5.8: Hypercycle Estimates for the Obstacle Problem

Estimate	Level 0	Level 1	Level 2	Level 3	Level 4
$\mathcal{H}^{(\kappa)}(x_h, \rho_0)$	1.11	1.10	1.09	1.09	1.09
$\mathcal{H}^{(\kappa)}(x_h, \rho_0^{(10)})$	1.16	1.12	1.10	1.09	1.09
$\mathcal{H}^{(\kappa)}(x_h, \hat{\rho}_0)$	7.91	12.55	18.16	25.49	35.76
$\mathcal{H}^{(\kappa)}(x_h, \hat{\rho}_0^{(10)})$	1.28	1.42	1.78	2.12	2.61
$\mathcal{H}^{*}(x_h, \rho_0)$	3.63	3.86	6.06	7.03	10.64
$\mathcal{H}^{*}(x_h, \rho_0^{(10)})$	1.13	1.10	1.10	1.10	1.11
$\mathcal{H}^{*}(x_h, \hat{\rho}_0)$	8.44	13.33	19.57	27.55	39.23
$\mathcal{H}^{*}(x_h, \hat{\rho}_0^{(10)})$	1.18	1.16	1.20	1.29	1.45
$\mathcal{H}^{(\kappa)}(x_h, \sigma_1)$	1.02	1.01	1.00	1.00	1.00
$\mathcal{H}^{(\kappa)}(x_h, \sigma_1^{(10)})$	1.07	1.03	1.01	1.00	1.00
$\mathcal{H}^{(\kappa)}(x_h, \hat{\sigma}_1)$	8.92	14.71	20.93	29.57	41.68
$\mathcal{H}^{(\kappa)}(x_h, \hat{\sigma}_1^{(10)})$	1.24	1.52	1.94	2.26	2.80
$\mathcal{H}^{*}(x_h, \sigma_1)$	3.10	3.11	4.94	5.36	8.30
$\mathcal{H}^{*}(x_h, \sigma_1^{(10)})$	1.04	1.02	1.03	1.01	1.01
$\mathcal{H}^{*}(x_h, \hat{\sigma}_1)$	9.55	15.72	22.39	31.56	44.87
$\mathcal{H}^{*}(x_h, \hat{\sigma}_1^{(10)})$	1.16	1.34	1.31	1.26	1.31
$\mathcal{H}^{(\kappa)}(x_h, \rho_1)$	4.00	5.26	7.05	9.47	13.10
$\mathcal{H}^{(\kappa)}(x_h, \rho_1^{(10)})$	1.12	1.09	1.10	1.17	1.27
$\mathcal{H}^{(\kappa)}(x_h, \hat{\rho}_1)$	16.05	32.49	68.55	141.42	287.40
$\mathcal{H}^{(\kappa)}(x_h, \hat{\rho}_1^{(10)})$	2.62	3.39	4.78	7.47	14.14
$\mathcal{H}^{*}(x_h, \rho_1)$	7.59	10.31	14.64	20.22	29.04
$\mathcal{H}^{*}(x_h, \rho_1^{(10)})$	1.25	1.23	1.52	1.63	2.10
$\mathcal{H}^{*}(x_h, \hat{\rho}_1)$	19.00	39.01	82.75	171.01	347.87
$\mathcal{H}^{*}(x_h, \hat{\rho}_1^{(10)})$	1.90	2.46	3.75	5.60	9.14

The figure 5.8 shows, that the alternative hypercycle estimate, we have discussed in section 2.2.5, can be much more accurate than our analysis in section 3.3.2 warrants. We must keep in mind, however, that we are studying an example which is rather contrived: The boundary of the contact set is a manifold of lower dimension (a simple ring), whence the impact of the lacking regularity, which the analytic solution of the obstacle problem exibits but here, may be felt not that severely. Away from the free boundary either the obstacle function is contained in the primal ansatz, or the admissibility constraint can be satisfied exactly even in the lowest order case. Though the minimiser of the alternative hypercycle estimate $\mathcal{H}^{(\kappa)}$ has been identified as the gradient of the solution to the perturbed obstacle problem specified in proposition 2.1, the numerical solution of the original problem in its dual formulation yields a very good error bound, that can hardly be improved by subsequent minimisation steps. On the contrary, the results can even be worse, if we carry out a minimisation procedure without properly fixing the equilibration

parameter. The generic procedure described in proposition 1.1 delivers somewhat disappointing hypercycle estimates $\mathcal{H}^*(x_h, \cdot)$, which is in keeping with our deliberations in section 2.2.8. Only after the solution of the dual formulation has been modified by a minimisation procedure, the error bounds become comparable to the alternative hypercycle estimates. Speaking but in qualitative terms, the generic estimates derived from the numerical solution of the dual formulation are similar to those estimates of either type, which rely on the recovery of the gradient by means of an orthogonal projection. These latter estimates exhibit the asymptotic behaviour we would expect with a view to the limited regularity of the analytical solution. Indeed, a closer look at the tables C.2 and C.3 in the appendix C reveals, that the smoothed gradient fails to properly reduce the residual in the admissibility constraint. With respect ot the alternative hypercycle estimates $\mathcal{H}^{(\kappa)}(x_h, \cdot)$ we note, that a fixed number of minimisation sweeps is insufficient to remedy the shortcomings of the smoothed gradient. The generic error estimates, however, respond much better to such a post-processing scheme.

Figure 5.9: Alternative Estimates for the Obstacle Problem

Estimate	Level 0	Level 1	Level 2	Level 3	Level 4
$\mathcal{R}_{h,\varepsilon}(x_h)$	6.58	6.74	6.85	6.92	6.96
$\mathcal{H}^{(\kappa)}(x_h, \tau_h)$	6.55	9.12	12.21	16.82	23.33
$\mathcal{H}^{(\kappa)}(x_h, \tau_h^{(10)})$	1.23	1.52	1.73	1.95	2.15
$\mathcal{H}^*(x_h, \tau_h)$	6.83	9.28	12.81	17.75	25.13
$\mathcal{H}^*(x_h, \tau^{(10)})$	1.85	1.75	2.63	2.88	3.64
$\mathcal{H}^*_{-1}(x_h, \rho_0)$	1.11	1.09	1.09	1.09	1.09
$\mathcal{H}^*_{-1}(x_h, \sigma_1)$	1.02	1.01	1.00	1.00	1.00
$\mathcal{H}^*_{-1}(x_h, \rho_1)$	1.08	1.04	1.02	1.01	1.00

Though Raviart-Thomas elements of higher order should, at least in principle, provide us with error estimates, that are superior to those obtained from lower order discretisations of the dual parameter, our experiments indicate the contrary. Our findings are difficult to explain: The minimiser of the hypercycle estimate $\mathcal{H}^{(\kappa)}$ is not identical with the solution of the obstacle problem in its dual formulation. This discrepancy may be felt the more acutely, the more accurate our ansatz is. However, the generic estimates $\mathcal{H}^*(x_h, \cdot)$ suffer from the use of higher order trial spaces as well. Consulting the appendix C we see, that the poor performance of the estimates is caused by the residuals $M_R^{(\kappa)}$ and M_R^* in the second duality relation. It almost seems, as if the greater latitude, the higher order elements give as in dealing with these residuals, was the cause for the difficulties we encounter. Probably, we can improve our results by adapting the projection scheme, we have described in section 5.2.5, in basically the same way we have modified the computation of an upper bound for the hypercycle estimate (2.15). Unfortunately, pertinent experimental data is currently not available to test such a hypothesis.

The second figure 5.9 presents no overwhelming surprises. When we employ the product ansatz $(P_c^1(\mathcal{M}_h))^2$ to construct the dual parameter we sacrifice some flexibility while using practically the same shape functions the Brezzi-Douglas-Marini element offers. Therefore, we may expect such an ansatz to yield error estimates slightly worse than those obtained from Brezzi-Douglas-Marini elements and slightly better than those derived from Raviart-Thomas elements of lowest order. In fact, by projecting the gradient ∇x_h into the product space we obtain the best results for the alternative hypercycle estimate (2.19) without actually solving the dual formulation of the obstacle problem. The performance of the gradient thus smoothed is superior to the other recovery schemes in the generic case (2.16) as well. However, the continuous vector field seems to require more minimisation sweeps than any of the other two choices does. The sharpest error bounds are derived from the estimate (2.15) by the procedure outlined in section 5.3.3. Consulting table C.1 we find the output of the residual based error estimator somewhat disappointing, even if we ignore the seizable consistency error $\mathcal{S}_\epsilon$ and its impact on the efficiency index. With a view to figure 5.7 decreasing ϵ hardly seems to be a viable option in any case.

5.4 Error Estimates on Locally Refined Meshes

The numerical experiments, we have reported in the previous sections, indicate, that generalised hypercycle estimates can deliver very sharp bounds on the so called energy error $|\, x_h - x_0 \,|_{\Omega,1}$. Such high an accuracy comes at a price, however: The computational resources necessary to define, for instance, the dual parameter $\rho_1 \in R_c^1(\mathcal{M}_h)$ exceed by far those, we must dedicate to the calculation of the primal approximation $x_h \in P_c^1(\mathcal{M}_h)$. In table 5.10 we have compiled the dimensions of various vector spaces involved in the numerical experiments. Hereby, the function $\mu_i \in P^i(\mathcal{M}_h)$ designates the Lagrange multiplier for the kinematic admissibility constraint.

Figure 5.10: Numerical Complexity of various Hypercycle Estimates

	Level 0	Level 1	Level 2	Level 3	Level 4
x_h	433	1849	7633	31009	124993
τ_h	866	7872	31488	62018	249986
ρ_0	1536	6024	23856	94944	378816
σ_1	3072	12048	47712	189888	757632
ρ_1	5040	19920	79200	315840	1261440
μ_0	984	3936	15744	62976	251904
μ_1	2952	11808	47232	188928	755712

It seems evident, that a mesh adaption procedure should provide the very means to improve hypercycle estimates without overly increasing their computational costs. Below we will present a few numerical results, which have been obtained by refining the mesh $\mathcal{M}'_h$ used to resolve the dual parameter (see figure 5.1 for an explanation of the notation). In one set of tests the mesh $\mathcal{M}_h$ has been refined globally in each stage of the experiment. Thus the efficiency indices of the resulting hypercycle estimates can be compared directly against those we have presented in the previous sections. In a second set of tests the mesh $\mathcal{M}_h$ has stayed unchanged. In that way the impact of the local mesh refinement on the efficiency of the error bounds becomes more apparent.

5.4.1 Error Estimates for the Laplace Problem

To guide the local mesh refinement we use the following simple heuristics: For each element $M \in \mathcal{M}'_h$ within the mesh let us compute its contribution to the second part $M_R^{(\kappa)}$ of the hypercycle estimate defined by (2.8). Hereby, we must choose the value of the equilibration parameter $\kappa > 0$ as we see fit, for its optimal value κ^* can only be determined a posteriori by (5.2):

$$M_R^{(\kappa)}(\tau, M) \quad := \quad \frac{1+\kappa}{\lambda_0^2} \parallel \nabla\cdot \tau + f \parallel_M^2 \qquad ; \qquad \tau \in H_{\mathrm{div}}(\Omega) \ .$$

We proceed by sorting the elements in descending order with respect to this local indicator. Having fixed a certain quota $\alpha \in (0,1)$ beforehand we may now schedule the first $\lfloor \alpha\, N \rfloor$ elements of the mesh for refinement, if the total number of elements within the mesh amounts to N. By the above procedure, also known as *fixed fraction strategy* we may hope to reduce the impact of the second component $M_R^{(\kappa)}$ on the final hypercycle estimate $\mathcal{H}^*$ and to obtain an approximation of the gradient ∇x_0, that is more suitable for the data of the problem.

In all of our experiments we have used the setting $\kappa = 0.5$ to determine those elements, we want to refine. Thereby, the dual parameter $\tau'_h \in R_c^0(\mathcal{M}'_h)$ has been computed as the numerical solution of the dual formulation (3.40). The general structure of the simulation code is outlined in section 5.1. The results of the mesh adaption procedure are illustrated in figure 5.16 for the example C. In this very case, the refinement parameter reads $\alpha = 0.1$.

Both tables 5.11 and 5.12 demonstrate, that the use of mesh refinement can be beneficial to the resulting hypercycle estimates. In the case of the latter figure, the advantages of the mesh

Figure 5.11: Efficiency Indices for Locally Refined Meshes

α	Example	Level 0	Level 1	Level 2	Level 3	Level 4
0.05	A	7.24	5.69	4.36	3.57	3.07
0.10	A	7.24	5.68	4.34	3.48	2.82
0.15	A	7.24	5.66	4.24	3.45	2.82
0.20	A	7.24	5.65	4.30	3.45	2.73
0.30	A	7.24	5.64	4.28	3.44	2.71
1.00	A	7.24	5.63	4.26	3.41	2.68
0.05	B	3.00	2.59	2.24	2.07	1.96
0.10	B	3.00	2.36	2.07	1.89	1.72
0.15	B	3.00	2.20	1.93	1.70	1.57
0.20	B	3.00	2.10	1.81	1.59	1.48
0.30	B	3.00	2.06	1.70	1.51	1.36
1.00	B	3.00	1.95	1.46	1.23	1.11
0.05	C	6.43	4.42	3.64	3.09	2.80
0.10	C	6.43	4.17	3.02	2.61	2.28
0.15	C	6.43	4.11	2.74	2.27	1.98
0.20	C	6.43	4.07	2.63	2.02	1.78
0.30	C	6.43	4.05	2.58	1.85	1.53
1.00	C	6.43	4.03	2.53	1.76	1.38

adaption are not that apparent. We must not forget, though, that the efficiency index of the energy error estimate should be multiplied roughly by a factor of two for each refinement level, we leave the mesh $\mathcal{M}'_h$ completely unchanged. There is a trade-off between the accuracy of the hypercycle estimate and the ratio of those resources we must spend on the solution of the primal and the dual formulation. If we can accept a deterioration of the error bounds in the course of successive refinements of the primal mesh $\mathcal{M}_h$, a moderate refinement of the mesh $\mathcal{M}'_h$ can lead to acceptable results, which stay extremely cheap. To give an example: The efficiency index of the hypercycle estimate $\mathcal{H}^*(x_h^{(C)}, \tau'_h)$ on refinement level 4 reads 12.66 for a refinement parameter $\alpha = 0.3$ and is not even twice as large as the estimate we find in figure 5.3. However, the latter estimate requires the solution of linear system involving 378816 degrees of freedom for the dual parameter ρ_0 alone, while the former involves but 29010 unknowns to define τ'_h.

If we are primarily interested a tool to guide the refinement of the mesh $\mathcal{M}_h$, computing dual parameters on coarser meshes is obviously a viable option. If we need accurate error bounds at all costs, we clearly have to calculate our a posteriori estimates with excess refinement of the mesh $\mathcal{M}'_h$. Figure 5.11 illustrates, that but little additional refinement may be sufficient to significantly improve the efficiency of the resulting hypercycle estimate. The impact of the excess refinement is most pronounced in the example A, where the discontinuity of the forcing function f must be resolved. Though the mesh $\mathcal{M}'_h$ supports but 5475 degrees of freedom on level 4, if we set $\alpha = 0.05$, the efficiency index of the corresponding hypercycle estimate is more than halved compared to the initial result, which requires 1536 degrees of freedom. We beg to note, that the lacking regularity of the analytical solution $x_0^{(A)}$ impairs the efficiency of the error bounds, even when the mesh $\mathcal{M}'_h$ is refined globally. The best result we can hope for reads:

$$I_{\text{eff}} \quad = \quad 1 \; + \; \mathcal{O}(\sqrt{h'}) \quad .$$

Hereby, h' denotes the local mesh parameter of $\mathcal{M}'_h$ in a neighbourhood around the discontinuity

Figure 5.12: Deterioration of I_{eff} due to Incomplete Refinement

α	Example	Level 0	Level 1	Level 2	Level 3	Level 4
0.05	A	7.24	10.69	15.38	24.05	38.79
0.10	A	7.24	10.45	14.74	22.32	33.69
0.15	A	7.24	10.37	14.63	22.02	32.81
0.20	A	7.24	10.35	14.33	21.35	30.93
0.30	A	7.24	10.33	14.15	20.77	29.51
0.05	B	3.00	4.39	6.85	11.68	20.99
0.10	B	3.00	3.92	5.92	9.82	16.97
0.15	B	3.00	3.60	5.36	8.64	14.65
0.20	B	3.00	3.39	4.99	7.91	13.39
0.30	B	3.00	3.31	4.55	7.15	11.58
0.05	C	6.43	8.03	12.61	21.73	36.91
0.10	C	6.43	7.48	9.72	16.72	29.00
0.15	C	6.43	7.32	8.46	13.01	23.17
0.20	C	6.43	7.20	7.82	10.41	17.74
0.30	C	6.43	7.17	7.55	8.70	12.66

of the function f as it has been described in section 5.2.1.

5.4.2 Error Estimates for the Obstacle Problem

Again, let us employ a *fixed fraction* strategy to adapt the mesh $\mathcal{M}_h'$ to the data of the obstacle problem, we have specified in section 5.2.1. As a local indicator we introduce the quantity:

$$M_R^{(\kappa)}(x_h, \tau, M) \quad := \quad \frac{1+\kappa}{\lambda_0^2} \left\{ \left\| \nabla\cdot \tau + f \right\|_M^2 \; - \; \left\| \, 0 \wedge \left(\lambda_0^2 \, \frac{x_h - \psi}{1+\kappa} \; + \; \nabla\cdot \tau \; + \; f \right) \right\|_M^2 \right\}$$

which we define on any element $M \in \mathcal{M}_h'$ for any vector field $\tau \in H_{\text{div}}(\Omega)$. In the case of Laplace problem, the dependence of the local indicators $M_R^{(\kappa)}(\cdot, M)$ on the equilibration parameter does not raise any difficulties. Since all the local indicators are merely scaled by the same amount, the sort sequence of the elements is not affected by our choice of κ. In the case of the obstacle problem the situation is clearly different, as the local weights $M_R^{(\kappa)}$ depend on the equilibration parameter κ in a nonlinear fashion. The best results could possibly be obtained by running the algorithm described in figure 5.6 to determine the optimal equilibration parameter κ^*. Once this quantity would be known, the weights $M_R^*(x_h, \cdot, M)$ should be computed and the elements sorted in ascending order with respect to the local indicator. The first $\lfloor \alpha \, N \rfloor$ elements were to be refined, if N designates the total number of elements in the mesh $\mathcal{M}_h'$.

For simplicity, we have fixed $\kappa = 10$ in all the experiments reported below. The parameter $\tau_h' \in R_c^0(\mathcal{M}_h')$ is determined with the help of the mixed discretisation scheme, we have introduced in section 4.3.2. The algorithm we use to solve the linear complementary problem (4.13) has been detailed in section 4.3.4. Again, figure 5.16 may be consulted to learn the general structure of the simulation code. The computation of the final hypercycle estimate is described in section 5.3.3, while the post-processing of the parameter τ_h' is performed with the help of the relaxation procedure we have discussed in section 5.2.4. The performance of the resulting mesh adaption procedure is illustrated in figure 5.17. The refinement parameter reads $\alpha = 0.1$.

In figure 5.13 the results of two sets of experiments have been collected. In the first set of tests the local weights $M_R^{(\kappa)}(\cdot, \cdot, M)$ have been used. The mesh adaption strategy thus defined

Figure 5.13: Efficiency Indices for Locally Refined Meshes

	α	Level 0	Level 1	Level 2	Level 3	Level 4
$\mathcal{H}^{(\kappa)}$	0.05	1.11	1.08	1.08	1.08	1.07
	0.10	1.11	1.08	1.07	1.06	1.06
	0.15	1.11	1.07	1.05	1.04	1.03
	0.20	1.11	1.07	1.04	1.03	1.02
	0.30	1.11	1.06	1.03	1.02	1.02
$\mathcal{H}^*$	0.05	1.13	1.10	1.11	1.13	1.44
	0.10	1.13	1.09	1.10	1.15	1.90
	0.15	1.13	1.09	1.11	1.25	2.36
	0.20	1.13	1.08	1.13	1.45	2.79
	0.30	1.13	1.08	1.17	1.69	3.27

aims at improving the alternative hypercycle estimate derived in [41]. To indicate this fact the corresponding rows in the table have been tagged with the label $\mathcal{H}^{(\kappa)}$. The second set of experiments covers the hypercycle estimate we have derived in section 2.2.4. Accordingly, those rows of the table, which pertain to the estimate (2.16), are labelled by $\mathcal{H}^*$. All the experiments have been conducted in the same manner except for the definition of the local indicators, which determine the sort sequence and thence the subset of elements to be refined. In the latter set of experiments the following indicators have been employed:

$$M_R^*(\tau, M) \quad := \quad \| \nabla\cdot \tau + f \|_M^2 \; - \; \| 0 \wedge (\nabla\cdot \tau + f) \|_{M\cap\Omega_x}^2 \quad ; \qquad \tau \in H_{\mathrm{div}}(\Omega) \ .$$

Hereby, $\Omega_x \subset \Omega$ denotes the *coincidence set* of the numerical solution x_h.

Figure 5.14: Deterioration of I_{eff} due to Incomplete Refinement

	α	Level 0	Level 1	Level 2	Level 3	Level 4
$\mathcal{H}^{(\kappa)}$	0.05	1.11	1.26	1.72	2.74	4.95
	0.10	1.11	1.24	1.61	2.19	3.06
	0.15	1.11	1.22	1.52	1.96	2.78
	0.20	1.11	1.20	1.42	1.71	2.46
	0.30	1.11	1.16	1.26	1.50	1.95
$\mathcal{H}^*$	0.05	1.13	1.33	1.85	2.84	5.06
	0.10	1.13	1.26	1.61	2.44	4.43
	0.15	1.13	1.23	1.52	2.25	4.03
	0.20	1.13	1.18	1.45	2.10	3.68
	0.30	1.13	1.15	1.31	1.67	2.23

Figure 5.14 indicates, that the impact of successive refinements of the mesh $\mathcal{M}_h$, on which we compute the primal approximation x_h, can be compensated for to a reasonable extent by a local refinement strategy for the "dual" mesh $\mathcal{M}_h'$. Those remarks we have made with a view to the table 5.12 clearly apply also here: The trade-off between the computational resources, we must spent on the calculation of the hypercycle estimate, and the resulting accuracy seems to be balanced more in the favour of a local mesh adaption, however. The reason for this is probably the presence of the coincidence set, on which with the numerical solution x_h coincides with the *flat* obstacle ψ. Since x_h can be resolved exactly with trial functions from the ansatz $P_c^1(\mathcal{M}_h)$ even on the coarsest mesh, the mesh adaption should take place mainly outside the coincidence

set. In consequence, the number of elements affected by a refinement procedure is significantly reduced and even moderate refinement quota are rendered effective.

Figure 5.15: Numerical Complexity and Local Mesh Refinement

	α	Level 0	Level 1	Level 2	Level 3	Level 4
τ'_h	0.05	1536	1968	2421	3072	3849
	0.10	1536	2283	3336	4836	7128
	0.15	1536	2565	4152	6699	10791
	0.20	1536	2884	5339	9307	16141
	0.30	1536	3512	8118	17243	36179
μ'_h	0.05	984	1272	1574	2008	2526
	0.10	984	1482	2184	3184	4712
	0.15	984	1670	2728	4426	7154
	0.20	984	1882	3515	6160	10716
	0.30	984	2294	5345	11417	24019

The lacking regularity of the analytical solutions to those constrained variational problems, which define the minimisers of the hypercycle estimates $\mathcal{H}^{(\kappa)}$ and $\mathcal{H}^*$, impairs the accuracy of those error bounds, we can actually obtain from dual parameters in some finite dimensional trial space. The results put forth in the propositions 3.12 and 3.13 are clearly very pessimistic. Their proof (as well as the material in the section 2.2.7) indicates, that the hypercycle estimates can benefit substantially from a local mesh refinement around the boundary of the coincidence set. The mesh parameter prominent in both propositions is actually but a local one related to those elements of the mesh, in which the regularity of the primal solution breaks down.

Our numerical experiments corroborate this notation, as the figure 5.17 illustrates. The table 5.13 contains a somewhat puzzling summary of our results: The hypercycle estimate $\mathcal{H}^{(\kappa)}$ discussed in section 2.2.5 exhibits exactly the behaviour we should expect with a view to our deliberations in section 3.3.2. The generic a posteriori estimate $\mathcal{H}^*$ however suffers from the refinement of the mesh $\mathcal{M}'_h$. Since the effect is not felt immediately, we must presume, that the solution of the obstacle problem in its dual formulation on the adapted mesh $\mathcal{M}'_h$ no longer yields a good approximation for the minimiser of the hypercycle estimate. Moreover, a fixed number of relaxation sweeps (4 in this case) seems the less appropriate to remedy this problem, the more degrees of freedom are involved. Our findings are collected in the tables C.20 and C.21. Though it has not been documented in these tables, we would like to add, that the alternative error bound $\mathcal{H}^{(\kappa)}$ has proved much less reliant on the post-processing than its generic counterpart.

Figure 5.16: Local Mesh Refinement for the Model Problem C

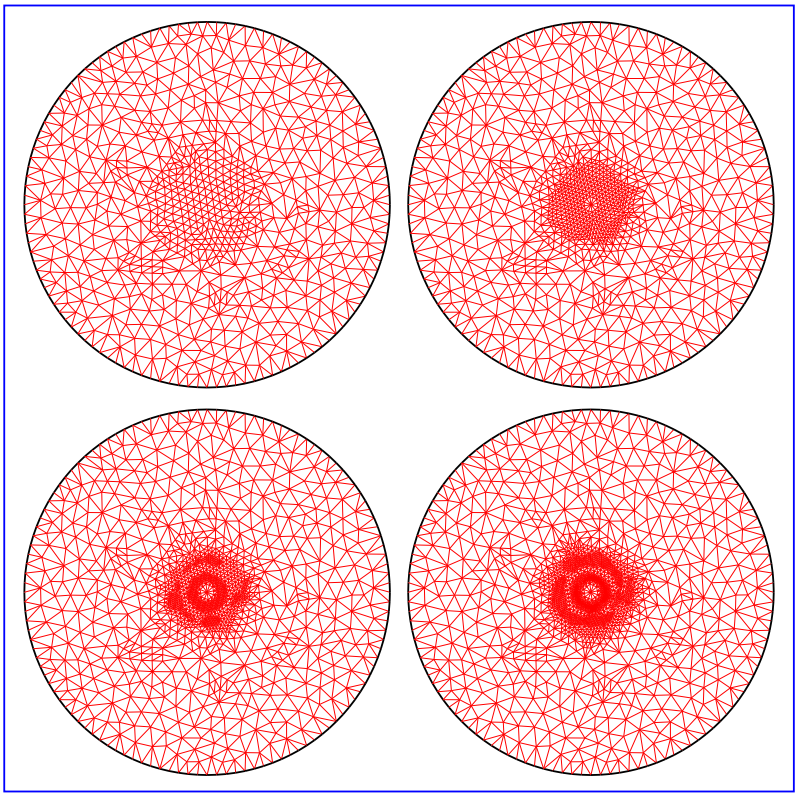
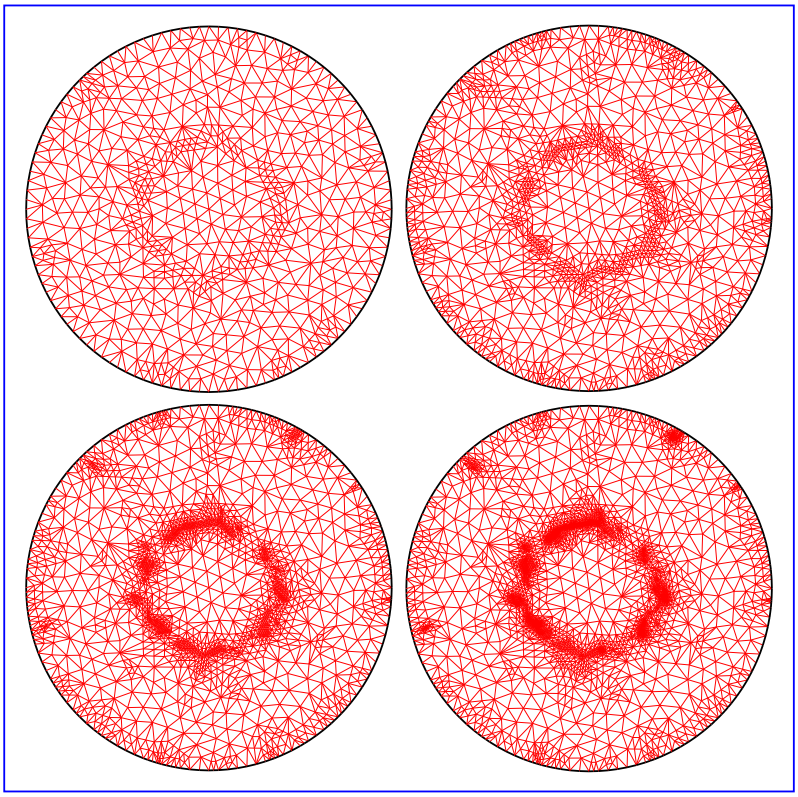


Figure 5.17: Local Mesh Refinement for the Obstacle Problem



Conclusion

In the introduction to this text we have outlined the principal mechanisms, which practically all a posteriori error estimators currently in use rely on. Hence, we shall not repeat here the implicit assumptions on the data of the problem, which are necessary for their successful application, or their inherent limitations. By employing analytical techniques from the calculus of variations we have developed in chapter 1 a more general approach to the computation of a posteriori error estimates, which covers a wide range of applications: We have focused on those problems

$$F(\Lambda x) \; + \; G(x) \qquad \underset{x \in X}{\longrightarrow} \qquad \min$$

involving a uniformly convex functional $F : Y \longrightarrow \mathbb{R}$ and a coercive linear operator $\Lambda : X \longrightarrow Y$, a setting first explored probably by Rockafellar [127]. The key to the successful calculation of bounds on the approximation error $\| \Lambda x_h - \Lambda x_0 \|_Y$ is the formulation of a dual problem, which supplies lower bounds on the minimal "energy" that we may associate with the solution $x_0 \in X$ of the primal formulation. We have focused on this point of view exclusively, though there are a number of fields (mechanics for instance), in which the dual statement

$$- \; F^*(y^*) \; - \; G^*(-\Lambda^* y^*) \qquad \underset{y^* \in Y^*}{\longrightarrow} \qquad \max$$

may be the more natural choice to consider. However, the analytical framework we present in chapter 1 proves to be symmetric with respect to the primal and the dual formulation. Hence, our methodology enables us to simultaneously devise a posteriori error estimates both for a primal discretisation scheme and its dual counterpart. In fact, the a posteriori estimators we propose can be identified in certain generic cases as dual formulations of the original problems, augmented though by some lower order perturbation terms: The dual formulation of a certain Helmholtz problem for instance delivers an error bound for the Dirichlet problem and the error estimator for the obstacle problem proves related to a constrained variational formulation of Helmholtz type. The introduction of lower order perturbations has already been discussed in the pioneering work of Babuška and Rheinboldt [16]. While their approach seems ad hoc, our analysis succeeds in putting their ostensibly technical device into a broader perspective.

In section 1.3 we have employed an augmented Lagrangian to extend our analytical framework and derive bounds on functional outputs. Though our deliberations have been limited to the study of quadratic forms, our findings can still be regarded as a significant progress, if we compare them against known results [116, 121]. Our analysis is easily adapted to the case of either convex or concave functionals, if we care but for one-sided bounds on their output. In addition, we do not require any of the quantities involved in the computation of the bounds to satisfy certain auxiliary problems *exactly*. Therefore, these bounds are reliable despite the fact, that we usually employ numerical cubature schemes to assemble the data of the variational formulation and that we do not solve the discrete problems up to machine precision. A window of inaccuracy, so to speak, is introduced merely by the numerical procedure, which we evaluate the bound with. Under those assumptions, which are prerequisite for the more established analytical methods to be applicable, we find the numerical expense comparable, which we must invest into the computation of our estimates: An auxiliary problem must be solved to find the preimage of the output under the action of the elliptic operator, we may associate with the quadratic form F. Subsequently, some recovery procedure is necessary to determine a dual parameter related to the preimage.

The analysis of two applications in chapter 2 has exemplified, how the energy error estimates, we have obtained with the help of duality techniques from the calculus of variations, bear on both residual based error estimators and conventional hypercycle estimates. We have seen, that the key to the efficiency of those error bounds, we have termed "generalised hypercycle estimates", lies in the proper choice of certain dual parameters. If we were to create them at will, the generalised hypercycle estimates would be exact. For all practical purposes, however, we must construct them in suitable trial spaces. Our numerical experiments have explored basically three approaches, how to find the dual parameters: By exploiting the duality mapping, by solving

the dual formulation of the original problem and by minimising the hypercycle estimates with respect to the dual parameters. We can infer from our numerical data, that the second method, possibly combined with the third, delivers the most satisfactory results. With a view to the close ties between the a posteriori error estimator and the dual formulation this could be expected. Surprisingly, though, we have also found recovery schemes to perform worst, which project the image of the numerical solution under the action of the duality mapping into a conforming ansatz.

The latter result is unfortunate, since recovery schemes can be implemented easily and require little computational resources. Speedily computing the solution of the dual formulation, may be difficult however. To illustrate our point let us but remark, that only two papers [39, 111], the latter of which has been written by the author, seem to have appeared so far, which deal with multilevel solvers for the obstacle problem in a dual setting. Several multilevel methods (e. g. [31]) have been proposed for the unconstrained problem, which exploit the fact, that the dual mixed discretisation is equivalent to a nonconforming discretisation scheme for the primal formulation [8, 9]. In [39] we have discussed three different possibilities, how to compute the solution of saddle point problems with constraints on the Lagrange multipliers. Meanwhile, we have found a proof of convergence for a smoothing scheme first presented in said paper. The proof along with some other material on the design of fast multilevel methods has been rendered in section 4.3. The FAS scheme presented in section 4.3.6 has been found to perform best, if Raviart-Thomas elements of lowest order are used to set up the dual mixed discretisation (see [39]). The FAS scheme presupposes, that the continuity of the normal fluxes associated with the solution of the dual formulation is enforced in a weak sense with the help of Lagrange multipliers. Since these multipliers may be extended into the whole of the computational domain with the help of Crouzeix-Raviart elements, the method can be understood as a generalisation of those multigrid methods, which deal with the nonconforming discretisation of the primal formulation and transform the results back into the dual setting.

It seems intriguing to employ hybrid discretisation schemes in order to compute the solution of the primal and the dual formulation *simultaneously*. The generalised hypercycle estimates do not really require a conforming dual parameter. However, the evaluation of upper bounds on certain dual norms can only be avoided, if we assume some additional degree of regularity. The primal approximation must be admissible in any case, as else there would be no a priori bound on the approximation error in terms of the functional $F \circ \Lambda + G$. This puts us in a difficult position: If we relied on conventional hybrid finite elements, the numerical solution of the primal formulation would be nonconforming and the hypercycle estimate pointless. In chapter 3 we have discussed the ramifications of using generalised hypercycle estimates in a finite element context. Section 3.2 is devoted to the analysis of an alternative hybridisation procedure, which delivers *conforming* primal approximations. Numerical experiments, which illustrate the performance of this novel approach, have been published in [40]. Unfortunately, the dual solution is nonconforming, which we obtain thanks to the new discretisation technique. To obtain computable a posteriori error estimates the resort outlined in section 5.2.5 must be employed.

A different approach to reducing the computational overhead connected with the evaluation of generalised hypercycle estimates consists in the use of locally refined meshes, on which the dual parameters are computed. The bulk of chapter 4 has been dedicated to technical issues: Among other topics we have discussed the management of locally refined meshes, the insertion of special refinement patterns and the construction of suitable mesh transfer operators. Thereby, we have used the code of our finite element libraries to exemplify our concepts. The numerical techniques we have implemented do not only apply to the computation of a posteriori error estimators: they can also be profitably employed, whenever data obtained on one locally refined mesh must be transferred onto another one.

Whether to implement hypercycle estimates as a means of guiding the local mesh refinement or a means to asses the accuracy with which certain quantities have been computed, is an awkward question to raise. The answer must depend on the capabilities of the software tools employed as well as the requirements of the application. If the reliability of the error estimate and its accuracy are of paramount importance, duality techniques can hardly be avoided. If the error estimator is primarily used to control a mesh refinement procedure, the quality of the estimates must be balanced against the computational resources spent on finding them. In such a scenario recovery schemes may perform best, even if their theoretical foundation is violated. The technical effort necessary to compute generalised hypercycle estimates in a time efficient manner can be

substantial. If the numerical tools support trial spaces of Raviart-Thomas type and support hybrid discretisation schemes, only little additional programming may be mandatory. If this is not the case, the implementation of the technical framework can prove prohibitively expensive. Below, we give a summary statement of the advantages and of the drawbacks, we feel connected with the use of generalised hypercycle estimates:

a) Generalised hypercycle estimates are applicable to a wide range of variational problems under minimal assumptions on the data. Constrained and unconstrained problems can be analysed within the same framework.

b) The underlying mathematical theory does not require the numerical solution to satisfy any best approximation property. Therefore, the following situations can be dealt with rigorously: i) The data of the problem necessitates the use of some inexact numerical cubature scheme. ii) The algebraic problem has not been solved exactly, as the iterative process, designed to obtain the solution, had to be terminated prematurely. iii) The numerical solution has not been computed with some finite element method at all.

c) Generalised hypercycle estimates do not rely on the existence of any stability estimates, which relate the approximation error to the data of the problem.

d) The generalised hypercycle estimates consist of two contributions: While one is directly related to the energy error, the other may be seen as a penalty term on the consistency of the data. The first and more relevant contribution involves no unknown interpolation constants. The efficiency of the hypercycle estimate is therefore but little affected by modifications of the mesh and the "calibration" of the error estimator is unnecessary.

e) Depending on the choice of certain parameters, generalised hypercycle estimates can prove extremely sharp. There are no inherent limitations to their accuracy but for the computational resources, which must be spent finding said parameters.

f) The information provided by the hypercycle estimates is easily turned into a criterion which a local mesh adaption strategy can be based on.

g) Using suitably defined hybrid discretisation schemes the numerical solution of the primal formulation and the generalised hypercycle estimate can be computed simultaneously.

h) As long as the numerical solution of the primal problem is admissible, the generalised hypercycle estimate is always reliable.

The above list of advantages is balanced to some extent by the following disadvantages:

a) Simple procedures to determine those "dual" parameters, which affect the accuracy of the hypercycle estimates, usually result in comparatively poor error bounds.

b) To realise their full potential generalised hypercycle estimates must rely on the solution of some suitable defined dual formulation. Efficient numerical procedures for that task may not be available or at least may not be that well understood.

c) The treatment of the dual formulation usually leads to additional technical requirements, which must be met by the finite element software employed for the solution of the primal problem. The programming is rather involved, especially for locally adapted meshes.

Unresolved Problems and Future Challenges

Perhaps, no disquisition of any import can ever aspire to be exhaustive. While this thesis has managed to discover a new principle on the strength of which a posteriori error estimators for a certain class of variational problems can be derived in a generic manner, it has also raised a number of interesting questions, which unfortunately have been left unanswered. The most prominent among these questions, is certainly a problem connected with the material presented in section 4.3: To the best of our knowledge there is still no proof, why multigrid methods for constrained variational problems deliver contraction rates uniformly bounded in the mesh

parameter h. Another important problem, we have failed to address, is the design of multilevel methods for hybrid, mixed discretisation schemes, if the index k of the Raviart-Thomas space $R^k(\mathcal{M}_h)$ is odd. This is the reason, why we have used the more general, but definitely slower, approach for our own numerical experiments of working with the Schur complement and solving for the Lagrange multipliers of the admissibility constraints.

In section 3.2 we have developed a new discretisation scheme for the dual mixed formulation of elliptic differential equations. The Lagrange multipliers for the continuity constraints of the normal fluxes can be used to compute a *conforming* approximation to the solution of the primal formulation. Unfortunately, we have not been able to assert, whether this approximation is of optimal order, when the index of the Raviart-Thomas space $R^k(\mathcal{M}_h)$ is odd. The problem is related to the design of multilevel methods for hybrid discretisation schemes: it seems impossible to properly extend these Lagrange multipliers into the interior of the finite elements. Our analysis of the approximation properties has been carried out in the natural norms. It would be helpful to know, whether similar results could also be obtained in the L^∞-norm. The event would justify our using the quantity $M_D(x_h, \cdot)$ as a local error indicator.

Our deliberations in section 3.2 pertain but to the unconstrained variational problem. We have not attempted to extend our analysis to the case of variational inequalities. As yet, such an analysis, carried out in the natural norms, seems only available for dual mixed discretisation schemes based on the standard Raviart-Thomas elements [37]. Hence, there are two questions still open: i) What are the approximation properties of Raviart-Thomas elements, if we measure them in the L^∞-norm? ii) Can we eventually translate our findings from conventional discretisation schemes to those nonconforming methods we have discussed in section 3.2?

In section 1.3 we have discussed ways, how to apply our analytical techniques to the definition of reliable bounds on functional outputs. Our results have been limited to minimisation problems involving quadratic forms. It is fairly obvious, that our methodology can also be used to derive either upper or lower bounds on functional outputs, if the functionals are convex respectively concave. The generalisation to convex variational problems has failed, however. We may look for possible extensions of our theoretical framework in two directions: i) Using suitable defined Lagrangians we may hope to develop reliable a posteriori error estimators for functional outputs, that are valid for a wider class of variational problems. ii) The theory may also be applicable to variational problems involving uniform quasi-convex functionals $F : Y \longrightarrow \mathbb{R}$.

Appendix A

A reference for the grammar of the FEM language

The language, whose grammar we are going to present, has been employed as an integral part of our implementation of the finite element method not only to define some general features of certain finite elements, but also to prescribe the behaviour of these elements under mesh transformations. We have designed a parser that can attain 482 different states and is aware of 56 terminal and 85 nonterminal symbols. These are related in accordance with 184 rules. It is therefore very tempting to simplify the presentation by leaving out some syntactical elements of minor importance. However, we see fit to take the opportunity and turn the following sections into a concise reference. Consequently, we will have to refer to some grammatical features, that seem but poorly accounted for with a view to our reasonings in the sections 4.2.2 and 4.2.3. If necessary, we will give a short explanation why certain key words or certain rules have been introduced into the language.

A.1 A note on the Backus-Naur Form

In the following we will repeatedly have to describe syntactical elements of a language, that can suitably characterise all relevant properties of a finite element scheme. To simplify this task we adopt a formalism, widely known as *BNF* notation. The acronym translates into *Backus Naur Form* and refers to a formal notation, introduced by J. Backus [17] in 1959 and finalised by P. Naur [110] a year later. This formalism comprises three meta-symbols and two types of symbols: The first type denotes grammatical rules, while the second one denotes grammatical primitives, that can be found verbatim in any text sample of the language under consideration. Symbols of the first type are usually called *nonterminal*, since they can be resolved into smaller syntactic groupings, whereas those of the second type are referred to as *terminal* symbols. The meta-symbols are used to designate definitions and logical disjunctions. The third meta-symbol serves to distinguish between terminal and nonterminal symbols. All grammars, that can be described in the BNF notation, are context free. However, not all context free grammars are easy to parse: The established tools for generating parsers are optimised for only a special subset, called *LALR(1)* grammars. Languages, that have such a grammar, can be parsed by deterministic bottom up techniques utilising a somewhat restricted look ahead of one symbol in order to collapse the state tables of the parser. (For background information on parsing technology we refer to: [77, 100].) Below we will use a slightly extended variant of the original Backus Naur formalism, which employs two additional meta-symbols to denote optional syntactic elements and to describe recurrences:

`::=`	denotes a definition by the right hand side
`|`	denotes an exclusive logical disjunction ("or")
`<...>`	surround category identifiers ("nonterminals")
`[...]`	embrace optional expressions
`{...}`	are used to abbreviate recursive rule definitions
`" "`	distinguish between single characters and meta-symbols

Remark A.1 The introduction of the additional meta-symbols `[...]` and `{...}` does not pose any difficulties, since any grammar, which we can describe with their help, can be easily expressed without using these meta-symbols at all. We may define optional expressions using additional disjunctions. Repetitive symbols we can incorporate into the grammar by introducing recursively defined rules. To give an example, let us consider the rule

```
<Text>  ::=  <Letter> { <Letter> | <Mark> }  .
```

We can resolve this definition recursively in the following manner:

```
<Text>    ::=    <Letter>
             |    <Text> <Letter>
             |    <Text> <Mark>    .
```

In a similar fashion the simple disjunction

```
<Name>   ::=   <FirstName> <LastName>
           |    <FirstName> <Patronymic> <LastName>
```

is nothing but the translation of the more compact notation

```
<Name>  ::=   <FirstName> [ <Patronymic> ] <LastName>  .
```

Remark A.2 The grammatical primitives of a language must be identified by pattern matching for the parser to decide which reductions to apply next. The process of finding these terminal symbols is called *lexical analysis*. Computer programs, which perform the necessary pattern recognition, are usually referred to as *scanners*. There are a number of tools available to generate scanners automatically. Hence, we will not follow up on this topic below. It is worth mentioning, however, that the scanner itself utilises a state table and can therefore be abused to process and modify the input stream, before tokens are generated and transmitted to the parser. Among the tasks, a scanner can perform independently of the parser, there are the removal of comments or whitespace from the input, the handling of nested input streams or the management of symbol tables. (See e. g. [3, 100] for references.)

A.2 The Description of the Grammar

We have already mentioned in remark A.2, that the process of parsing a text sample of some given language is intertwined with a process of pattern matching, usually termed scanning or lexical analysis. Hence, the shift or reduce operations, performed by the state engine within the parser, are not directly controlled by the data of the input stream, but by a sequence of *tokens*, usually but bare numbers fed to the parser by the scanner. These tokens may simply designate certain terminal symbols, such as reserved words for instance. They may also indicate however, that data of some kind has been retrieved from the input stream. The identification of such data and its subsequent retrieval are handled by the scanner in a transparent fashion. In our rule definitions we will have to use both types of tokens. While we will render all reserved words of our "FEM language" verbatim, there are a number of terminal symbols, we can not describe as easily. Such tokens we indicate by using uppercase letters only. Below we give a summary of these tokens and explain their meaning:

`EOL`	This token represents the newline ASCII character (`'\n'`).
`FLOAT`	This token indicates, that a floating point number has been read and stored.
`NUMBER`	An unsigned integer has been encountered and stored.
`SIGNED`	A signed integer value has been found.
`STRING`	A string has been detected and a handle deposited to be used by the parser.

Since the scanning of a stream of data for certain terminal symbols can be viewed as parsing a language with a regular grammar, the scanner is able to cope with a number of syntactic constructions on its own. The most common application is the removal of comment from the input. Another task, that can be handled transparently by the scanner, is the management of nested input files. Switching between different sources of data is facilitated by the "`Include`" statement. Its syntax reads:

`Include` : *NameOfFile* `EOL`

Comments can be placed anywhere in the input stream except immediately after the "`Include`" keyword. These comments may span several lines, if they are delimited in a C-style fashion:

`/*` *Text of comment...* `*/` .

In addition, short comments, which extend to the end of the current line only, can be commenced by either of the following two expressions:

`//` *Text of comment...* `EOL`

`#` *Text of comment...* `EOL` .

The topology of a finite element mesh is stored most conveniently along with all the information necessary to reconstruct the sequence of mesh transformations, which have been carried out to obtain the final mesh. Hence, the external representation of a mesh comprises four parts in total: A header, containing a description of all the finite elements, that constitute the mesh, a section, in which all the vertices of the mesh are defined, a section, in which these vertices are associated to finite elements, and eventually a section, in which the tree structure of the refinement hierarchy is determined. Hence, the top level rule reads:

```
<GridFile>  ::=  <ElementDescriptions> <VertexSection>
                 <ElementSection> <TopologySection>  .
```

We proceed and resolve the nonterminals in this rule:

```
<ElementDescriptions>  ::=  BeginSection Descriptors EOL                (A.1)
                            <DescriptorList>
                            EndSection EOL  ,

<VertexSection>  ::=  BeginSection Nodes EOL
                      <VertexEntry> { <VertexEntry> }
                      EndSection EOL  ,

<ElementSection>  ::=  BeginSection Elements EOL
                       <ElementEntry> { <ElementEntry> }
                       EndSection EOL  ,

<TopologySection>  ::=  BeginSection Topology EOL
                        <TopologyEntry> { <TopologyEntry> }
                        EndSection EOL  .
```

Strictly speaking, it would be far from necessary to introduce another nonterminal symbol in rule (A.1). The structure of this definition could be same as that of the following three ones. However, an additional definition of the form

```
<DescriptorList>  ::=  <Descriptor> { <Descriptor> }                    (A.2)
```

is very convenient as an alternative top level rule, if only element properties are to be specified and the mesh data is imported from an external source. Below we will discuss the nonterminal symbols, we encounter in the rules (A.1) up to (A.2). We will begin our examination with the symbol most easily resolved and work our way to the more difficult cases:

```
<VertexEntry>  ::=  NUMBER ":" FLOAT { FLOAT } EOL  .
```

The number in front of the separator serves to identify the vertex in the subsequent section of the grid file, in which the elements are to be defined. In a strict sense, the separator is unnecessary: By its presence the readability of a grid specification can nevertheless be enhanced, if such a specification has to be drawn up by hand for testing purposes. The following list of floating point numbers specifies the cartesian coordinates of the vertex. The length of each list must be the same, otherwise the parser stops and signals an error condition. The length of the list is used to determine the dimension of the domain under consideration:

```
<ElementEntry>  ::=  NUMBER ":" STRING ":" <ElementKey> ":"
                     <ListOfNodes> EOL
                |    NUMBER ":" STRING ":" <ElementKey> ":"
                     NUMBER ":" NUMBER ":" <ListOfNodes> EOL   .
```

The leading number is used in the topology section of the grid file to furnish a reference for the element, while the following string is employed to uniquely assign a finite element descriptor from among the list of descriptors, specified in the first section of the input. For reasons explained below **the element index must be positive**. The key of the element, a signed integer quantity, is at our disposal to identify certain parts of the mesh, e. g. to distinguish between various boundary conditions on certain parts of the boundary. If further numbers follow, the first of them denotes the type of refinement, the element has been subjected to, while the second one denotes the status of a finite automaton, which is employed in the generation of surface grids. If these numbers are missing, both the refinement type and the state of the automaton are switched to *unrefined*. Finally, all vertices of the element are listed by their vertex numbers:

```
<ListOfNodes>  ::=  NUMBER { NUMBER }   .
```

The length of this list must conform to the element specification as it has been designated by the string. The order, in which the different vertices are enumerated, is used to determine the spatial orientation of the patch. The first number corresponds to the first vertex of the reference patch, the second number to the second vertex and so forth. On the purpose of the element key we have already commented:

```
<ElementKey>  ::=  NUMBER | SIGNED   .
```

An entry in the topology section of the input is slightly more involved, as a number of different situations can be encountered: The element may be contained in the bottom layer of the mesh, it may be contained in the surface layer or it can be found somewhere in-between. In the latter case we have to specify the root as well as the leaf elements to reconstruct the refinement hierarchy. In one of the other two cases we can drop at least one piece of information:

```
<TopologyEntry>  ::=  NUMBER ":" <ListOfNeighbours>
                      [ ":" NUMBER ] EOL
                 |    NUMBER ":" <ListOfNeighbours> ":"
                      [ NUMBER ] ":" <ListOfLeaves> EOL   .
```

The leading number refers to the current element. The following list denotes the adjoining elements, that can be found on the same level within the refinement hierarchy. If a neighbour is missing, this is indicated by using the fictitious element index `"0"`. In the element section, it is therefore not allowed to assign this index to an actual finite element! Unless the bottom layer has been reached, the optional number in-between separators denotes the index of the root element, while the second list specifies, if present, the indices of the leaf elements to be found on the next higher refinement level:

```
<ListOfLeaves>  ::=  NUMBER { NUMBER }

<ListOfNeighbours>  ::=  NUMBER { NUMBER } .
```

These two rules more or less conclude the specification of the FEM grammar, as far as the external representation of hierarchically refined meshes is concerned. However, the header of any mesh description must contain detailed information, which kind of finite elements are to be fitted into the prescribed geometry, and how these elements are supposed to interact with each other in case of further mesh transformations. Hence, the bulk of our task still lies ahead of us. As a means of structuring an element description, all the necessary information is distributed among several dedicated subsections. Accordingly, we resolve the nonterminal `Descriptor` as:

```
<Descriptor>  ::=  BeginElement: STRING EOL
                   [ SetupProcedure: STRING EOL ]
                   UCD_Code: STRING EOL
                   <ListOfSubSections> EndElement EOL  .
```
(A.3)

The first string of characters is used for referencing purposes within the third part of the input stream, that is the `<ElementSection>`, as well as in the header itself. The optional second string can be used to override the internal finite element setup mechanism of the finite element package, since this has been designed to handle isoparametric Lagrange elements only. However, if specialised setup algorithms are required, a *SetupProcedure* handle can be passed to the FEM library as a drop in. To determine the proper replacement, the optional string is matched against the return value of the member function *SetupProcedure::IdentifyYourself()* for all handles, that are registered with the library. If the optional part of the rule is missing completely, the element is treated as a Lagrange element. The third string of characters is employed by various filters, that process output for external finite element tools. If data is generated in the widely recognised *Unstructured Cell Data* format, this string for example is used verbatim to designate the cell type. The list of subsections is defined recursively:

```
<ListOfSubSections>  ::=  <SubSection>
                     |    <ListOfSubSections> <SubSection>  .
```

This list of subsections must comprise a segment, which the quadrature rule to be employed on the reference patch $\hat{\Omega}$ is specified in, a segment describing the geometry of this very domain, and according to definition 3.1 a segment, in which a number of maps $\{\vartheta_1, \dots \vartheta_K\} \subset W^{k,\infty}(\hat{\Omega})$ is defined, with the help of which the reference patch is mapped onto the computational domain:

```
<SubSection>  ::=  BeginSubSection: GaussPoints EOL
                   <Point> { <Point> } EndOfSubSection EOL
                |  BeginSubSection: Weights EOL
                   <Weight> { <Weight> } EndOfSubSection EOL
                |  BeginSubSection: Vertices
                   <Point> { <Point> } EndOfSubSection EOL
                |  BeginSubSection: DoFPoints EOL
                   <DoFPoint> { <DoFPoint> } EndOfSubSection EOL
                |  BeginSubSection: UCD_Map EOL
                   <Point> { <Point> } EndOfSubSection EOL
                |  BeginSubSection: Boundaries EOL
                   <FaceEntry> { <FaceEntry> }
                   EndOfSubSection EOL
```
(A.4)

```
|  BeginSubSection: FormFunctions EOL
   <FunctionString> { <FunctionString> }
   EndOfSubSection EOL
|  BeginSubSection: MapFunctions EOL
   <FunctionString> { <FunctionString> }
   EndOfSubSection EOL
|  BeginSubSection: Refinement EOL
   <RefinementScheme> { <RefinementScheme> }
   [ <TableOfRefinementSchemes> ]
   EndOfSubSection EOL   .
```

We proceed and discuss the various segments of an element description in more detail: The first subsection specifies the location of the cubature nodes within the reference patch with the help of *cartesian* coordinates. By definition the reference patch is convex, so natural coordinates are a viable alternative. However, there are no significant advantages in using natural coordinates instead of cartesian ones. The nonterminal `Point` is resolved in the following manner:

```
<Point>  ::=  NUMBER ":" FLOAT { ":" FLOAT } EOL  .
```

The first number is nothing but a counter for the number of quadrature points specified. As such is it present more or less for aesthetical reasons: The correspondence between cubature nodes and weights is established according to their sequence and not to their index. The following list of floating point numbers records the exact location of the cubature point within the reference patch. In a similar fashion the nonterminal `Weight` is resolved:

```
<Weight>  ::=  NUMBER ":" FLOAT EOL  .
```

The subsection `Vertices` serves to determine the reference patch, which is assumed to be the convex hull of all the points specified in this segment. While the indices of the cubature nodes are not meaningful by themselves, the enumeration of all vertices is requisite for referencing purposes in the `Boundaries` subsection. With a view to definition 3.1 there must exist exactly one vertex for each function $\vartheta_i : \hat{\Omega} \longrightarrow \mathbb{R}$ specified in the `MapFunctions` subsection.

```
<DoFPoint>  ::=  NUMBER [ ":" <DoFKey> ]
                 ":" FLOAT { ":" FLOAT } EOL  .
```

According to remark 4.3 each degree of freedom should be associated with a set of coordinates, even if this degree of freedom cannot be computed by a point evaluation. Establishing the correspondence between the various degrees of freedom and their respective coordinates is the purpose of the subsection `DoFPoints`. Each entry in this segment is constructed in a fashion similar to the nonterminal `Point`. However, to distinguish between distinct degrees of freedom, which share the same coordinates, we can assign a special key to them:

```
<DoFKey>  ::=  NUMBER | STRING  .
```

The use of strings as qualifiers is discouraged, as at present only a limited set of abbreviations is recognised by the parser! The matched strings are translated into numerical keys anyhow, as internally the finite element library works with numerical qualifiers only. For each entry in the subsection `DoFPoints` there must be either a corresponding entry in the segment `FormFunctions` or a fitting *SetupProcedure* handle registered with the library. Again, the enumeration of the entries is important for referencing the various degrees of freedom.

The `UCD_Map` segment facilitates the generation of output for external data processing tools. Since the degrees of freedom $\mathcal{S}$ of some trial space $\mathfrak{V}$ need not necessarily correspond to actual function

values attained at certain points within the computational domain Ω, it may become necessary to interpolate functions from the space $\mathfrak{V}$ with a set of suitably chosen polynomials. Using the coordinates $\{\hat{u}_1, \dots \hat{u}_K\} \subset \hat{\Omega}$, that are listed in this segment, a basis $\{\lambda_1, \dots \lambda_K\} \subseteq P_l$ of such polynomials is constructed for each element in the top layer of the mesh. Several output filters employ these polynomials to carry out a Lagrange interpolation in one or more spatial dimensions prior to exporting the data. Hence, the polynomials satisfy the relation:

$$\lambda_i(\hat{u}_j) \;=\; \delta_{ij} \qquad ; \qquad i,j \;\in\; \{1, \dots K\} \quad .$$

The subsection `Boundaries` serves to define the interfaces between finite elements: Each face of the finite element under consideration is identified with another finite element of lower spatial dimension. Thereby, the interface element is designated by the very label which has been assigned in rule (A.3). The geometry of the interface is specified by listing those of the vertices defined in the subsection `Vertices`, that span the hypersurface:

```
<FaceEntry>  ::=  NUMBER [ ":" STRING ] ":" <ListOfVertices> EOL  .
```

As it is not reasonable to view the endpoints of a line segment as finite elements in their own right, the definition of interfaces can be suppressed by omitting the label. However, if line segments are used to compute boundary integrals in a two-dimensional setting, these one-dimensional finite elements form a regular part of the mesh and therefore share interfaces with elements which belong to the interior of the domain: the interface of the line segment and the finite element itself coincide. Hence, in default of sensible alternatives this special scenario is assumed by the parser, whenever the label is missing. Consequently, the list of vertices must always be present:

```
<ListOfVertices>  ::=  NUMBER { NUMBER }   .
```

The next subsection is optional and must be absent from the finite element descriptor, if the finite element library has been provided with the appropriate *SetupDescriptor* handle. If a finite element of Lagrange type is to be specified, this segment serves to define the proper shape functions. The function definitions themselves are written down as strings terminated by the token `";"`:

```
<FunctionString>  ::=  <FunctionName> "=" <Expression> ";"   .
```

The name of the shape function is in itself meaningless. Nevertheless, the left hand side of the nonterminal `FunctionString` is necessary to identify the arguments in the body of the shape function. This very part of the function definition is resolved as:

```
<FunctionName>  ::=  STRING "(" STRING { "," STRING } ")"   .
```

The right hand side of the function definition is repeatedly evaluated by an interpreter in order to prepare certain tables, which are needed in the default setup procedure for isoparametric Lagrange elements. As such elements usually have only polynomial shape functions, the interpreter has but limited means at its disposal. To increase its speed, a symbol table has been dispensed with. Therefore, in fact no other but polynomial expressions can be computed by the current implementation. The grammar for the body of the shape function can be described by:

```
<Expression>  ::=  <Expression> "+" <Expression>
               |   <Expression> "-" <Expression>
               |   <Expression> "*" <Expression>
               |   <Expression> "/" <Expression>
               |   "-" <Expression>
               |   "(" <Expression> ")"
               |   FLOAT | STRING   .
```

The last rule is necessary to identify those arguments of the shape function, which have been defined while resolving the nonterminal `FunctionName`. The second but last rule matches any

floating point numbers, which are used explicitly in the body of the shape function. For the mandatory segment `MapFunctions` the same remarks apply.

The segment `Refinement` implements one or more strategies how a finite element is to behave under hierarchical mesh refinement. If only one refinement method has been specified, no internal state table is needed to control the generation of suitable surface meshes. In such a case, it is unnecessary to fix any set of rules to govern the element's state engine and we may suppress the dedicated part of the segment. We resolve the nonterminal `TableOfRefinementSchemes` as:

```
<TableOfRefinementSchemes>  ::=  <TableEntry> { <TableEntry> }  .
```

Each entry in the table consists of a leading string, which designates the refinement method to be applied, and a list of interfaces, the element must share with other, already refined elements, to be eligible for the type of refinement under consideration. The indices used to refer to particular interfaces correspond to those numbers specified in the `Boundaries` subsection. Additionally, a condition can be imposed on the state of the element itself. Manipulating the internal state of the element thus becomes a means of switching between different refinement strategies. The information when to apply a certain type of refinement we collect in the statement:

```
<RefinementKey>  ::=  NUMBER { "," NUMBER } [ State NUMBER ]
                  |   "-" { "-" } State NUMBER  .
```

The selection of the refinement method actually applied to a given finite element is based on finding the closest possible match from among the list of all refinement methods available. Hence, it can be reasonable to define a refinement procedure in default of any more appropriate method, that does not rely on any interface requirements. In a case like that the list of interfaces can be suppressed by inserting at least one `"-"` token. The specification of the refinement state becomes mandatory, as the default state *unrefined* is assumed, if no refinement state is specified for the element at all. To implement different policies for the generation of hierarchically refined grids, the formulation of alternative regular refinement procedures is an additional possibility. Following the `State` token the status of the finite element and optionally the status of the grid have to be specified. These conditions are to be matched exactly, before any alternative refinement method is applied. If the status of the grid is missing, a default state is substituted by the parser. The corresponding part of the grammar reads:

```
<RegularRefinement>  ::=  State NUMBER [ "(" NUMBER ")" ]  .
```

Since different geometrical constellations can necessitate the same auxiliary refinement procedure, the nonterminal `TableEntry` must actually involve lists of refinement criteria. It is also possible, that different states of the grid require the same strategy for regular grid refinement. Hence, in can also be important to assign several conditions for regular refinement to a certain refinement scheme. For technical reasons, however, such a scheme can never be of an auxiliary and of a regular character at the same time. To conclude our remarks, let us state that the nonterminal `TableEntry` is resolved in the following way:

```
<TableEntry>  ::=  STRING ":" <RegularRefinement>
                   { ":" <RegularRefinement> } EOL
               |   STRING ":" <RefinementKey>
                   { ":" <RefinementKey> } EOL   .
```

We proceed to discuss the specification of the various refinement procedures, we want to employ. Each of our schemes is identified by the very designation used in the table of refinement criteria as described by the above rule. Optionally, a number can be assigned to the refinement scheme, which reflects the internal state of the finite element requisite for the particular refinement method to be applied. If no number and hence no constraint on the element state is specified, the choice

of the refinement scheme is exclusively based on geometrical considerations:

```
<RefinementScheme>  ::=  BeginRefinementScheme: STRING EOL
                         [ AssignedRefinementState: NUMBER ]
                         <RefinementSection> { <RefinementSection> }
                         EndRefinementScheme EOL
```

The body of the refinement descriptor consists of a list of sections, in one of which the assignment of the various degrees of freedom to the different interfaces of the finite element is detailed. The other sections describe the topology of the refinement pattern, respectively the geometrical properties of the newly created leaf elements:

```
<RefinementSection>  ::=  BeginSubsection: DoFScheme EOL
                          <DoFData> { <DoFData> }
                          EndSubsection EOL
                      |   BeginSubsection: NodeScheme EOL
                          <NodeData> { <NodeData> }
                          EndSubsection EOL
                      |   BeginSubsection:  Topology EOL
                          <TopologyData> { <TopologyData> }
                          EndSubsection EOL
                      |   BeginSubsection: StateTable EOL
                          <StateEntry> { <StateEntry> }
                          EndSubsection EOL   .
```
(A.5)

Neither of these three sections must be absent. A fourth optional segment can be used to equip the element with a finite automaton and to define the switching patterns of the state engine. Though an input stream is syntactically correct, if one of the mandatory sections is missing, the input will be not pass a consistency check by the parser and the FEM library will terminate. In case the fourth subsection has not been supplied, the library will assume, that the state of the each element scheduled for regular refinement is to be propagated through the refinement tree.

```
<DoFData>  ::=  BeginLeaf NUMBER EOL
                <DoFDataItem> { <DoFDataItem> }
                EndLeaf EOL   .
```
(A.6)

The information contained within each segment of the rule (A.5) except for the fourth one is spread across several subsegments, which correspond to the various leaf elements generated by the refinement procedure to be defined. Each of the leaves is identified by the number, that marks the position of this leaf in the `NodeScheme` segment. The bulk of the nonterminal `DoFData` contains specifications, how to deal with the degrees of freedom, that will come into existence when the element is split. In that process, we are repeatedly faced with one of four possible scenarios: i) The degree of freedom under consideration is identical with a degree of freedom, that belongs to the refined element. ii) The degree of freedom does not yet exist and has to be allocated. iii) The degree of freedom can be identified with another degree of freedom previously created while processing leaves with a lower index. iv) The degree of freedom belongs to one of the interfaces, the refined element shares with other elements in the mesh. All four possibilities

are covered by the following syntactical constructions:

```
<DoFDataItem>  ::=  NUMBER ":" NUMBER EOL
                 |  NUMBER ":" NUMBER ":" NUMBER EOL
                 |  NUMBER ":" New EOL
                 |  NUMBER ":" <ExportKey> EOL   .
```

The leading number references the degree of freedom under consideration and matches the index, that has been assigned in the `DoFPoints` segment of rule (A.4). If the first pattern is identified, the following number is used to designate the corresponding degree of freedom, that has been inherited from the root element. If the keyword `New` is encountered, a new degree of freedom is allocated and assigned to the leaf element. If two numbers follow, the first one identifies the leaf and the second one the actual degree of freedom, that is identical to the degree of freedom under consideration. This mechanism can only be applied, if the leaf has already been processed, such that the designated degree of freedom is guaranteed to exist. The fourth alternative is used to identify those degrees of freedom, that are located on one of the various interfaces. To cover this case, the nonterminal `ExportKey` is resolved as:

```
<ExportKey>  ::=  Export: NUMBER "(" NUMBER ")"
                  { "," NUMBER "(" NUMBER ")" }   .
```

The first number following the keyword `Export` designates the interface and matches one of those indices, that have been introduced in the `Boundaries` subsection. As the interface is a finite element itself, there exists an enumeration of all its degrees of freedom as specified in the `DoFPoints` segment of the rule (A.4). It may be used to establish the correct alignment of the element and the interface: accordingly, the second number in parenthesis denotes the index of the degree of freedom viewed as belonging solely to the interface. If the degree of freedom is associated with several interfaces, a situation that will only occur in three or more dimensions, additional interface specifications can be appended.

```
<NodeData>  ::=  BeginLeaf NUMBER ":" STRING [ State: NUMBER ] EOL
                 <NodeDataItem> { <NodeDataItem> }
                 EndLeaf EOL    .
```

Basically everything, that has been explained with a view to (A.6), also holds for the nonterminal `NodeData`. A minor modification consists of the fact, that the enumeration of all vertices within a given interface is now defined in the `Vertices` subsegment instead of the `Boundaries` segment. The string following the leaf index is employed to identify the correct finite element description of this very leaf. In such a manner refinement procedures can be defined, where the leaves are of a different type than their root element. Optionally, the default state of the newly created leaf may be specified. If that statement is missing, the state engine of the leaf will be initialised with the current state of the refined element. In either case the final state of the leaf is determined in accordance with the rules set forth in the `StateTable` segment of (A.5). Since the location of newly created vertices cannot be inferred from the geometry of the refined element, it is necessary to specify the coordinates of each new vertex with respect to the reference patch. Consequently, the nonterminal `NodeDataItem` is resolved as:

```
<NodeDataItem>  ::=  NUMBER ":" NUMBER EOL
                   |  NUMBER ":" NUMBER ":" NUMBER EOL
                   |  NUMBER ":" FLOAT { FLOAT } EOL
                   |  NUMBER ":" FLOAT { FLOAT } <ExportKey> EOL   .
```

The information collected in the segment `Topology` of the rule (A.5) is used to determine, which elements actually adjoin. Since an interface between two elements can be viewed as an element itself, we may consider the external interfaces of a leaf element as the leaves of interfaces, which

belong to its root. However, only such elements are presumed to have a common interface, that feature the same refinement level: auxiliary refinement patterns may therefore cause some leaf elements to actually be without adjoining elements on some of their faces.

```
<TopologyData>  ::=  BeginLeaf NUMBER EOL
                     <TopDataItem> { <TopDataItem> }
                     EndLeaf EOL
```

Accordingly, the nonterminal `TopDataItem` must account for three possibilities: i) The adjoining element at the specified interface is a previously created leaf of the same root, the element under consideration has been derived from. ii) The interface is of an external nature and can be identified with a certain leaf of an interface element as it has been specified in the `Boundaries` segment of rule (A.4). iii) There is no adjoining element on the same mesh level, as the leaf belongs to an auxiliary refinement pattern.

```
<TopDataItem>  ::=  NUMBER ":" NUMBER EOL
                |   NUMBER ":" <ExportKey> EOL
                |   NUMBER ":" NoExport EOL   .
```

Though the syntax has not changed, the semantics of the nonterminal `ExportKey` is somewhat different in the present context: Following the keyword `Export` exactly one pair of numbers is expected. The first index designates the interface of the refined element, as it has been specified in the `Boundaries` segment of rule (A.4). The second index defines which of the hypothetical leaf elements of said interface corresponds to the external interface of the leaf element under consideration. The keyword `NoExport` signals, that the interface specified by the preceding index does not have a matching neighbour of the same refinement level as the leaf element.

The only nonterminal, which we have not yet resolved, is the symbol `StateEntry`, which is used to define those finite automata whose impact on the mesh refinement we have discussed in the section 4.2. In figure 4.13 we have already presented two examples, how these automata may be specified. Each line in the state table starts with the keyword `State` followed by an index which designates the very state the element must be in for the rule to be applicable. The second index enclosed in parentheses defines as a further prerequisite the state of the mesh, which must be matched as well. Hence, by switching the state of the mesh we can disable or enable sets of rules simultaneously. For each leaf created by executing the refinement scheme a state is defined that the leaf's state engine is initialised with. Following another field separator the switching rule is finalised by specifying the state the element is to assume, once the refinement pattern on top of it has been removed. We may summarise:

```
<StateEntry>  ::=  State: NUMBER "(" NUMBER ")" ":"
                   NUMBER { "," NUMBER } ":" NUMBER EOL   .
```

Appendix B

An Algorithm for Performing Top Layer Refinements

To fully comprehend the source code of the method `FemGrid::PerformTopLayerRefinement` we render in figure B.2 some knowledge of the data structures is requisite which are internally used by our finite element library. The following paragraphs are intended as a primer and should by no means be considered as a technical reference.

General information

Each element in a mesh can be classified according to its geometrical properties and the shape functions it supports. Most of the manipulations involving a finite elements are so generic, that their execution depends only of the element's type and the coordinates of the element's vertices. Replicating the pertinent data which is necessary to carry out these manipulations in each element is therefore unnecessary and, moreover, too costly in terms of core memory consumption. Instead a pointer to an instance of a data structure termed `ElementDescriptor` has been added to the base class `ElementParameters`. Among other things said data structures each contain an array of unsigned integers termed `StateTable`, another array of unsigned integers called `RefinementMethod` and an array of data structures of the type `SplitDescriptor`.

Each entry in the array `StateTable` consists of unsigned integer, with the help of which the proper refinement method is identified. The structure of these keys will be discussed in a dedicated section. The second array `RefinementMethod` holds an index for the corresponding refinement scheme. The index is used to encode the current refinement pattern and to access the proper set of auxiliary data through the method `FemElement::GrabSplitDescriptor`. When a refinement pattern is removed, the refinement state of the former root element is switched. The new state depends on the refinement patters as well as the current state of the element. To account for the latter an array termed `StateKeyArray` forms a part of the data structure `SplitDescriptor`. The current refinement state is matched against the entries in this array. If two refinement keys match, the new state of the root element can be obtained from the corresponding entry in the array `SplitDescriptor::RootStateOnRemoval`. Each state of an element corresponds to a certain refinement pattern. To find this pattern the key `SplitDescriptor::RefinementState` must be compared against the supplied state identifier.

The composition of a refinement key

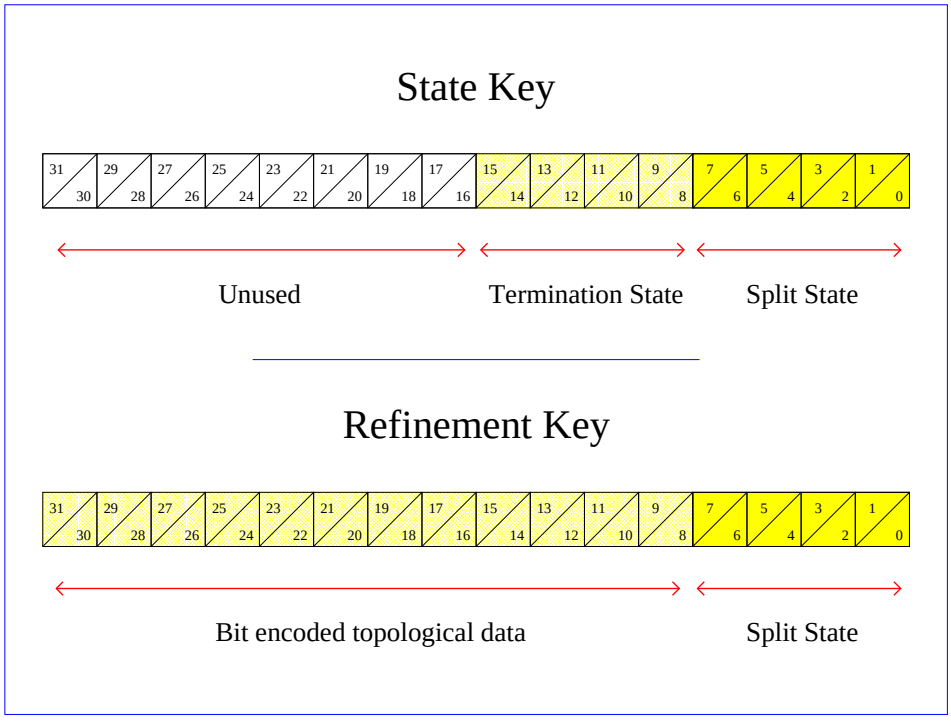
The refinement key has been implemented as an unsigned integer consisting of four bytes. The eight least significant bits of this number hold the element state, as it has been obtained through the method `FemElement::GetSpliteState`. The remaining bits encode the topology of the mesh: If the adjoining element with the index i has been refined the bit with the number $i + 8$ is set; if the adjoining element is unrefined, said corresponding is cleared. Due to this key layout our finite element library is limited to those elements with no more than 24 interfaces. Furthermore, we may not account for more than 256 states in the element's state engine. While the first restriction is not unduly severe, the latter may cause our implementation to fail in a tree-dimensional setting, since each refinement pattern is associated with an unique state. The refinement keys are stored in the array `ElementDescriptor::StateTable` in ascending order. Thereby, the very first keys in the table do not correspond to special refinement patterns. They are reserved for alternative regular refinement schemes. The first auxiliary scheme is indexed by the variable `ElementDescriptor::ExtraRefinement`, while the total number of keys stored in the table is indicated by the variable `ElementDescriptor::SizeOfStateTable`. Since the key table is sorted, the layout of the keys imposes a block structure: each group of keys corresponds to a certain geometrical constellation, while subsequent keys within one block correspond to the various internal states the root element can assume. Those refinement schemes, which are prompted by the element's state alone, are encoded by keys whose most significant bits are all

cleared. As such keys constitute the first block, they can be screened by setting the value of the variable `ElementDescriptor::ExtraRefinement` appropriately. Hence, the design of our keys is consistent with the layout of the key table.

The composition of a state key

As already mentioned, each refinement scheme can be identified by a unique state, which is stored in the variable `SplitDescriptor::RefinementState`. This state is encoded by an unsigned integer and consists of two parts. The eight least significant bits encode the refinement state, as it can obtained by calling the method `FemElement::GetSplitState`. The following bits 8 through 15 render the *termination state* of the mesh, which may be queried with the method `FemGrid::ReadTerminationState`. Since the termination state is a property of the mesh, while the refinement state is a property of each element, it is possible to maintain several coexisting refinement strategies for the same mesh. For instance, we may define a one on two refinement strategy as depicted in figure 4.10 and a one on three strategy simultaneously. Having refined the mesh several times with one of these strategies we can then modify the state of the mesh and thus switch to the other strategy. As the state engines of the individual elements are not affected by this change, the mesh refinement can proceed without further adjustments on the application level. At present, only two termination states have been specified, which carry the hexadecimal codes `0x00` and `0x01` respectively. If the least significant bit is set, auxiliary refinement patterns are employed. If the bit has been cleared, the finite element library employs static condensation to avoid the explicit treatment of hanging nodes (for a short discussion of the underlying principles we refer to the sections 3.1.1 and 4.1.4). The most significant bits of each state key are unused.

Figure B.1: Composition of state and refinement keys



```
void FemGrid::PerformTopLayerRefinement( void )
{
  unsigned TerminationState( ReadTerminationState() << 8 ) ;
  List<FemElement> TodoList ;
  unsigned n( HighestActiveLevel ) ;          // If the grid consists of
  if ( n == 0 )  return ;                     // only the bottom layer no
  while ( n > 0 )                             // auxiliar refinement is
  {                                           // necessary at all
    TodoList.Clear() ;
    unsigned k( 0 ) ;
    unsigned kmax = LevelData[ -- n ] -> EndOfLevel ;
    if ( n > 0 )  k = LevelData[ n - 1 ] -> EndOfLevel ;
    for ( ; k < kmax ; k++ )
    {
      ElementArray[k] -> FlagSetupMode() ;        // BEWARE: Setup mode
      TodoList.Append( ElementArray[k] ) ;        // flag is abused!
    }
    int RedRefinement = 1 ;
    while ( RedRefinement )
    {
      RedRefinement = 0 ;
      FemElement *E0 = TodoList.GotoStart() ;
      while ( E0 != NULL )
      {
        ElementDescriptor *D0 = E0 -> Descriptor ;
        unsigned *StateTable = D0 -> StateTable ;
        unsigned BitCode = 0 ;
        k = D0 -> NumberOfNeighbours ;
        while ( k > 0 )
        {
          BitCode <<= 1 ;
          Element *EN = E0 -> Neighbour[ -- k ] ;
          if ( EN == NULL || EN -> Leaf == NULL )  continue ;
          BitCode |= 0x1 ;
        }
        BitCode <<= 8 ;                           // Merge the current
        BitCode |= E0 -> GetSplitState() ;        // element status
        if ( BitCode )
        {                                         // Do refinement!
          kmax = D0 -> SizeOfStateTable ;
          k = D0 -> ExtraRefinement ;
          for ( ; k < kmax && StateTable[k] <= BitCode ; k++ ) ;
          if ( k > D0 -> ExtraRefinement )
          {
            unsigned RefCode = StateTable[ -- k ] ;
            RefCode ^= BitCode ;
            if ( ! ( RefCode & ~0xFF ) )
            {                                     // Test whether irre-
              k = D0 -> RefinementMethod[k] ;     // gular refinement
              E0 -> SetSplitScheme( k ) ;         // is feasible to
              E0 = TodoList.GetNext() ;           // eliminate hanging
              goto NextElement ;                  // nodes
            }
          }
          if ( BitCode & ~0xFF )     // Refinement necessary at all?
          {
            RedRefinement = 1 ;                   // Carry out regular
            E0 -> Split( ) ;                      // refinement
            kmax = D0 -> NumberOfNeighbours ;
            for ( k = 0 ; k < kmax ; k++ )
            {
              FemElement *EN = E0 -> Neighbour[k] ;
              if ( EN == NULL || EN -> TestSetupMode() )  continue ;
              if ( EN -> Leaf == NULL )
              {
                EN -> FlagSetupMode() ;       // Inspect all recently
                TodoList.Append( EN ) ;       // generated elements
              }
            }
          }
        }
        E0 -> ClearSetupMode() ;              // Remove all elements that
        TodoList.DeleteEntry() ;              // have been refined or do
        E0 = TodoList.GetEntry() ;            // not need any refinement
        NextElement: ;                        // at all
      }
    }
    FemElement *E0 = TodoList.GotoStart() ;       // Determine true
    for ( ; E0 != NULL ; E0 = TodoList.GetNext() ) // refinement
    {                                             // state of all
      unsigned l( TerminationState ) ;            // top layer ele-
      l |= E0 -> GetSplitState() ;                // ments
      unsigned k = E0 -> GetSplitScheme() ;
      SplitDescriptor &SD = E0 -> Descriptor -> RefineData[k] ;
      for ( k = 0 ; k < SD.SizeOfStateArrays ; k++ )
      {
        if ( l == SD.StateKeyArray[k] )
        {
          E0 -> SetSplitState( SD.RootStateOnRemoval[k] ) ;
          break ;
        }                                         // Update refinement state of
      }                                           // all top layer elements
    }
  }
  unsigned ArrayLength( NumberOfActiveElements ) ;
  FemElement **Array = CopyArray( ArrayLength, ElementArray ) ;
  for ( unsigned n = 0 ; n < ArrayLength ; n++ )
  {
    FemElement *E0 = Array[ n ] ;                          // Account for
    unsigned AssignedState = E0 -> GetSplitState() ;       // all methods
    if ( AssignedState )                                   // being only
    {                                                      // auxiliary!
      AssignedState |= TerminationState ;
      for ( k = 0 ; k < kmax ; k++ )
      {
        SplitDescriptor &SD = E0 -> GrabSplitDescriptor( k ) ;
        if ( SD.RefinementState == AssignedState )
        {                                                  // Matching
          E0 -> SetSplitScheme( k ) ;                      // scheme
          E0 -> AssignedSplit( k ) ;                       // detected!
          break ;
        }
      }
    }
  }
  DeleteArray( Array ) ;
}
```

Figure B.2: An algorithm for performing top layer refinements

Appendix C

A Compilation of our Numerical Results

Figure C.1: Hypercycle estimates for the obstacle problem

Level	0	1	2	3	4
$\mid x_0 - x_h \mid_{\Omega,1}$	0.10725	0.05477	0.02763	0.01386	0.00694
$\mathcal{R}_{h,\varepsilon}(x_h)$	0.51343	0.26523	0.13531	0.06833	0.03432
$\mathcal{S}_{\varepsilon}(f,\psi)$	0.22321	0.11154	0.05604	0.02813	0.01409
$\mid \hat{x}_{\varepsilon} - x_{h,\varepsilon} \mid_{\Omega,1}$	0.11200	0.05590	0.02793	0.01394	0.00696
$\hat{I}_{\text{eff}}$	4.58	4.74	4.84	4.90	4.93
I_{eff}	6.58	6.74	6.85	6.92	6.96
$\mathcal{H}^{(\kappa)}(x_h,\tau_h)$	0.70216	0.49955	0.33738	0.23316	0.16188
κ	0.1687	0.1187	0.0877	0.0627	0.0446
$M_D^{(\kappa)}(x_h,\tau_h)$	7.117e-2	2.648e-2	9.173e-3	3.206e-3	1.119e-3
$M_R^{(\kappa)}(x_h,\tau_h)$	4.212e-1	2.231e-1	1.047e-1	5.115e-2	2.508e-2
I_{eff}	6.55	9.12	12.21	16.82	23.33
$\mathcal{H}^{(\kappa)}(x_h,\tau_h^{(10)})$	0.14206	0.08352	0.04788	0.02698	0.01495
κ	3.3723	1.9874	1.3944	1.0693	0.8713
$M_D^{(\kappa)}(x_h,\tau_h^{(10)})$	1.557e-2	4.641e-3	1.335e-3	3.761e-4	1.041e-4
$M_R^{(\kappa)}(x_h,\tau_h^{(10)})$	4.616e-3	2.335e-3	9.576e-4	3.517e-4	1.195e-4
I_{eff}	1.32	1.52	1.73	1.95	2.15
$\mathcal{H}^{*}(x_h,\tau_h)$	0.73209	0.50844	0.35383	0.24600	0.17439
κ^{*}	0.1398	0.1048	0.0771	0.0560	0.0397
$M_D^{*}(x_h,\tau_h)$	1.027e-2	2.809e-3	7.393e-4	1.891e-4	4.778e-5
$M_R^{*}(x_h,\tau_h)$	5.257e-1	2.557e-1	1.245e-1	6.033e-2	3.036e-2
I_{eff}	6.83	9.28	12.81	17.75	25.13
$\mathcal{H}^{*}(x_h,\tau_h^{(10)})$	0.19871	0.09577	0.07255	0.03985	0.02525
κ^{*}	0.7157	0.7477	0.4222	0.3762	0.2879
$M_D^{*}(x_h,\tau_h^{(10)})$	1.337e-2	3.289e-3	7.963e-4	1.969e-4	4.881e-5
$M_R^{*}(x_h,\tau_h^{(10)})$	2.611e-2	5.883e-3	4.468e-3	1.391e-3	5.888e-4
I_{eff}	1.85	1.75	2.63	2.88	3.64

Figure C.2: Hypercycle estimates based on Raviart-Thomas elements of lowest order

Level	0	1	2	3	4
$\mid x_0 - x_h \mid_{\Omega,1}$	0.10725	0.05477	0.02763	0.01386	0.00694
$\mathcal{H}^{(\kappa)}(x_h, \rho_0)$	0.11883	0.06002	0.03018	0.01512	0.00756
κ	59.26	167.77	289.63	576.41	1085.28
$M_D^{(\kappa)}(x_h, \rho_0)$	1.389e-2	3.581e-3	9.080e-4	2.282e-4	5.713e-5
$M_R^{(\kappa)}(x_h, \rho_0)$	2.343e-4	2.134e-5	3.135e-6	3.958e-7	5.264e-8
I_{eff}	1.11	1.10	1.09	1.09	1.09
$\mathcal{H}^{(\kappa)}(x_h, \rho_0^{(10)})$	0.12423	0.06124	0.03035	0.01515	0.00757
κ	14.57	34.36	102.82	221.24	471.85
$M_D^{(\kappa)}(x_h, \rho_0^{(10)})$	1.444e-2	3.645e-3	9.120e-4	2.285e-4	5.717e-5
$M_R^{(\kappa)}(x_h, \rho_0^{(10)})$	9.915e-4	1.061e-4	8.870e-6	1.033e-6	1.212e-7
I_{eff}	1.16	1.12	1.10	1.09	1.09
$\mathcal{H}^{(\kappa)}(x_h, \hat{\rho}_0)$	0.84824	0.68752	0.50179	0.35330	0.24812
κ	0.1478	0.0909	0.0624	0.0442	0.0312
$M_D^{(\kappa)}(x_h, \hat{\rho}_0)$	9.266e-2	3.938e-2	1.479e-2	5.279e-3	1.865e-3
$M_R^{(\kappa)}(x_h, \hat{\rho}_0)$	6.269e-1	4.333e-1	2.370e-1	1.195e-1	5.970e-2
I_{eff}	7.91	12.55	18.16	25.49	35.76
$\mathcal{H}^{(\kappa)}(x_h, \hat{\rho}_0^{(10)})$	0.13679	0.07775	0.04920	0.02941	0.01812
κ	7.1458	3.6850	1.6302	1.0644	0.7184
$M_D^{(\kappa)}(x_h, \hat{\rho}_0^{(10)})$	1.641e-2	4.754e-3	1.501e-3	4.461e-4	1.373e-4
$M_R^{(\kappa)}(x_h, \hat{\rho}_0^{(10)})$	2.297e-3	1.290e-3	9.205e-4	4.191e-4	1.911e-4
I_{eff}	1.28	1.42	1.78	2.12	2.61
$\mathcal{H}^*_{-1}(x_h, \rho_0)$	0.11890	0.05996	0.03018	0.01511	0.00756
$M_D^*(x_h, \rho_0)$	1.366e-2	3.560e-3	9.049e-4	2.278e-4	5.708e-5
$M^*_{-1}(x_h, \rho_0)$	4.807e-4	3.560e-5	5.734e-6	4.827e-7	7.113e-8
I_{eff}	1.11	1.09	1.09	1.09	1.09
$\mathcal{H}^*(x_h, \rho_0)$	0.38990	0.21140	0.16730	0.09740	0.07385
$M_D^*(x_h, \rho_0)$	1.366e-2	3.560e-3	9.049e-4	2.278e-4	5.708e-5
$M_R^*(x_h, \rho_0)$	1.384e-1	4.112e-2	2.708e-2	9.258e-3	5.397e-3
I_{eff}	3.63	3.86	6.06	7.03	10.64
$\mathcal{H}^*(x_h, \rho_0^{(10)})$	0.12150	0.06040	0.03050	0.01521	0.00768
$M_D^*(x_h, \rho_0^{(10)})$	1.449e-2	3.635e-3	9.191e-4	2.290e-4	5.727e-5
$M_R^*(x_h, \rho_0^{(10)})$	2.730e-4	1.272e-5	1.114e-5	2.466e-6	1.779e-6
I_{eff}	1.13	1.10	1.10	1.10	1.11
$\mathcal{H}^*(x_h, \hat{\rho}_0)$	0.90509	0.72980	0.54084	0.38186	0.27221
$M_D^*(x_h, \hat{\rho}_0)$	1.193e-2	3.280e-3	8.690e-4	2.233e-4	5.653e-5
$M_R^*(x_h, \hat{\rho}_0)$	8.073e-1	5.293e-1	2.916e-1	1.456e-1	7.404e-2
I_{eff}	8.44	13.33	19.57	27.55	39.23
$\mathcal{H}^*(x_h, \hat{\rho}_0^{(10)})$	0.12613	0.06354	0.03304	0.01795	0.01003
$M_D^*(x_h, \hat{\rho}_0^{(10)})$	1.517e-2	3.787e-3	9.397e-4	2.317e-4	5.762e-5
$M_R^*(x_h, \hat{\rho}_0^{(10)})$	7.366e-4	2.497e-4	1.517e-4	9.042e-5	4.305e-5
I_{eff}	1.18	1.16	1.20	1.29	1.45

Figure C.3: Hypercycle estimates based on Brezzi-Douglas-Marini elements

Level	0	1	2	3	4
$\lvert x_0 - x_h \rvert_{\Omega,1}$	0.10725	0.05477	0.02763	0.01386	0.00694
$\mathcal{H}^{(\kappa)}(x_h, \sigma_1)$	0.10926	0.05515	0.02773	0.01389	0.00695
κ	80.47	272.99	446.06	968.10	1865.02
$M_D^{(\kappa)}(x_h, \sigma_1)$	1.179e-2	3.030e-3	7.675e-4	1.926e-4	4.821e-5
$M_R^{(\kappa)}(x_h, \sigma_1)$	1.465e-4	1.110e-5	1.721e-6	1.990e-7	2.585e-8
I_{eff}	1.02	1.01	1.00	1.00	1.00
$\mathcal{H}^{(\kappa)}(x_h, \sigma_1^{(10)})$	0.11432	0.05634	0.02788	0.01391	0.00695
κ	15.49	36.97	125.31	307.06	754.26
$M_D^{(\kappa)}(x_h, \sigma_1^{(10)})$	1.228e-2	3.090e-3	7.709e-4	1.929e-4	4.823e-5
$M_R^{(\kappa)}(x_h, \sigma_1^{(10)})$	7.924e-4	8.358e-5	6.152e-6	6.283e-7	6.395e-8
I_{eff}	1.07	1.03	1.01	1.00	1.00
$\mathcal{H}^{(\kappa)}(x_h, \hat{\sigma}_1)$	0.95649	0.80582	0.57844	0.40984	0.28923
κ	0.1155	0.0692	0.0489	0.0346	0.0244
$M_D^{(\kappa)}(x_h, \hat{\sigma}_1)$	9.472e-2	4.202e-2	1.560e-2	5.614e-3	1.995e-3
$M_R^{(\kappa)}(x_h, \hat{\sigma}_1)$	8.202e-1	6.073e-1	3.190e-1	1.624e-1	8.166e-2
I_{eff}	8.92	14.71	20.93	29.57	41.68
$\mathcal{H}^{(\kappa)}(x_h, \hat{\sigma}_1^{(10)})$	0.13298	0.08306	0.05365	0.03127	0.01941
κ	7.8753	5.2851	1.7886	1.0310	0.6270
$M_D^{(\kappa)}(x_h, \hat{\sigma}_1^{(10)})$	1.569e-2	5.802e-3	1.846e-3	4.964e-4	1.451e-4
$M_R^{(\kappa)}(x_h, \hat{\sigma}_1^{(10)})$	1.992e-3	1.098e-3	1.032e-3	4.815e-4	2.314e-4
I_{eff}	1.24	1.52	1.94	2.26	2.80
$\mathcal{H}^*_{-1}(x_h, \sigma_1)$	0.10948	0.05515	0.02774	0.01388	0.00694
$M_D^*(x_h, \sigma_1)$	1.165e-2	3.019e-3	7.657e-4	1.924e-4	4.819e-5
$M^*_{-1}(x_h, \sigma_1)$	3.405e-4	2.241e-5	3.787e-6	2.747e-7	4.220e-8
I_{eff}	1.02	1.01	1.00	1.00	1.00
$\mathcal{H}^*(x_h, \sigma_1)$	0.33250	0.17040	0.13660	0.07434	0.05758
$M_D^*(x_h, \sigma_1)$	1.165e-2	3.019e-3	7.658e-4	1.924e-4	4.819e-5
$M_R^*(x_h, \sigma_1)$	9.890e-2	2.603e-2	1.791e-2	5.334e-3	3.267e-3
I_{eff}	3.10	3.11	4.94	5.36	8.30
$\mathcal{H}^*(x_h, \sigma_1^{(10)})$	0.11160	0.05598	0.02834	0.01400	0.00701
$M_D^*(x_h, \sigma_1^{(10)})$	1.231e-2	3.128e-3	8.005e-4	1.953e-4	4.865e-5
$M_R^*(x_h, \sigma_1^{(10)})$	1.495e-4	6.254e-6	2.770e-6	5.762e-7	5.493e-7
I_{eff}	1.04	1.02	1.03	1.01	1.01
$\mathcal{H}^*(x_h, \hat{\sigma}_1)$	1.02405	0.86120	0.61855	0.43750	0.31135
$M_D^*(x_h, \hat{\sigma}_1)$	9.806e-3	2.719e-3	7.275e-4	1.877e-4	4.760e-5
$M_R^*(x_h, \hat{\sigma}_1)$	1.039e-0	7.389e-1	3.819e-1	1.912e-1	9.689e-2
I_{eff}	9.55	15.72	22.39	31.56	44.87
$\mathcal{H}^*(x_h, \hat{\sigma}_1^{(10)})$	0.12437	0.07326	0.03626	0.01744	0.00910
$M_D^*(x_h, \hat{\sigma}_1^{(10)})$	1.506e-2	5.231e-3	1.240e-3	2.588e-4	5.700e-5
$M_R^*(x_h, \hat{\sigma}_1^{(10)})$	4.066e-4	1.350e-4	7.523e-5	4.524e-5	2.576e-5
I_{eff}	1.16	1.34	1.31	1.26	1.31

Figure C.4: Hypercycle estimates based on Raviart-Thomas elements of higher order

Level	0	1	2	3	4
$\mid x_0 - x_h \mid_{\Omega,1}$	0.10725	0.05477	0.02763	0.01386	0.00694
$\mathcal{H}^{(\kappa)}(x_h, \rho_1)$	0.42863	0.28786	0.19467	0.13123	0.09088
κ	0.3257	0.2326	0.1647	0.1179	0.0826
$M_D^{(\kappa)}(x_h, \rho_1)$	4.514e-2	1.564e-2	5.358e-3	1.816e-3	6.300e-4
$M_R^{(\kappa)}(x_h, \rho_1)$	1.386e-1	6.723e-2	3.254e-2	1.541e-2	7.628e-3
I_{eff}	4.00	5.26	7.05	9.47	13.10
$\mathcal{H}^{(\kappa)}(x_h, \rho_1^{(10)})$	0.12021	0.05943	0.03042	0.01621	0.00883
κ	7.8696	11.99	10.16	5.9442	3.6848
$M_D^{(\kappa)}(x_h, \rho_1^{(10)})$	1.282e-2	3.260e-3	8.422e-4	2.249e-4	6.127e-5
$M_R^{(\kappa)}(x_h, \rho_1^{(10)})$	1.629e-3	2.718e-4	8.291e-5	3.783e-5	1.663e-5
I_{eff}	1.12	1.09	1.10	1.17	1.27
$\mathcal{H}^{(\kappa)}(x_h, \hat{\rho}_1)$	1.72151	1.77917	1.89404	1.96040	1.99432
κ	0.0542	0.0269	0.0128	0.0062	0.0031
$M_D^{(\kappa)}(x_h, \hat{\rho}_1)$	1.523e-1	8.303e-2	4.535e-2	2.374e-2	1.214e-2
$M_R^{(\kappa)}(x_h, \hat{\rho}_1)$	2.811e-0	3.082e-0	3.542e-0	3.819e-0	3.965e-0
I_{eff}	16.05	32.49	68.55	141.42	287.40
$\mathcal{H}^{(\kappa)}(x_h, \hat{\rho}_1^{(10)})$	0.28083	0.18570	0.13200	0.10353	0.09809
κ	0.7324	0.4978	0.3112	0.1819	0.0879
$M_D^{(\kappa)}(x_h, \hat{\rho}_1^{(10)})$	3.334e-2	1.146e-2	4.136e-3	1.650e-3	7.772e-4
$M_R^{(\kappa)}(x_h, \hat{\rho}_1^{(10)})$	4.552e-2	2.302e-2	1.329e-2	9.069e-3	8.844e-3
I_{eff}	2.62	3.39	4.78	7.47	14.14
$\mathcal{H}_{-1}^{*}(x_h, \rho_1)$	0.11568	0.05673	0.02813	0.01398	0.00695
$M_D^{*}(x_h, \rho_1)$	1.111e-2	2.952e-3	7.586e-4	1.915e-4	4.524e-5
$M_{-1}^{*}(x_h, \rho_1)$	2.268e-3	2.666e-4	3.292e-5	4.061e-6	3.048e-6
I_{eff}	1.08	1.04	1.02	1.01	1.00
$\mathcal{H}^{*}(x_h, \rho_1)$	0.81430	0.56460	0.40440	0.28020	0.20150
$M_D^{*}(x_h, \rho_1)$	1.109e-2	2.950e-3	7.576e-4	1.914e-4	4.806e-5
$M_R^{*}(x_h, \rho_1)$	6.519e-1	3.158e-1	1.628e-1	7.834e-2	4.057e-2
I_{eff}	7.59	10.31	14.64	20.22	29.04
$\mathcal{H}^{*}(x_h, \rho_1^{(10)})$	0.13460	0.06763	0.04207	0.02261	0.01458
$M_D^{*}(x_h, \rho_1^{(10)})$	1.218e-2	3.099e-3	7.775e-4	1.938e-4	4.836e-5
$M_R^{*}(x_h, \rho_1^{(10)})$	5.941e-3	1.475e-3	9.928e-4	3.174e-4	1.641e-4
I_{eff}	1.25	1.23	1.52	1.63	2.10
$\mathcal{H}^{*}(x_h, \hat{\rho}_1)$	2.03790	2.13646	2.28664	2.37044	2.41391
$M_D^{*}(x_h, \hat{\rho}_1)$	7.824e-3	2.178e-3	5.732e-4	1.467e-4	3.707e-5
$M_R^{*}(x_h, \hat{\rho}_1)$	4.145e-0	4.562e-0	5.228e-0	5.619e-0	5.827e-0
I_{eff}	19.00	39.01	82.75	171.01	347.87
$\mathcal{H}^{*}(x_h, \hat{\rho}_1^{(10)})$	0.20409	0.13493	0.10359	0.07764	0.06339
$M_D^{*}(x_h, \hat{\rho}_1^{(10)})$	1.460e-2	3.820e-3	9.774e-4	2.506e-4	6.266e-5
$M_R^{*}(x_h, \hat{\rho}_1^{(10)})$	2.705e-2	1.439e-2	9.754e-3	5.778e-3	3.956e-3
I_{eff}	1.90	2.46	3.75	5.60	9.14

Figure C.5: Error estimates on uniformly refined meshes, Example A

Level	0	1	2	3	4
$\mid x_0 - x_h \mid_{\Omega,1}$	0.03938	0.02000	0.01015	0.00512	0.00256
$\mathcal{R}_h(x_h)$	0.16688	0.08781	0.04511	0.02288	0.01151
I_{eff}	4.24	4.39	4.44	4.47	4.50
$\mathcal{H}^*(x_h)$	0.37526	0.27898	0.19723	0.13971	0.09818
$M_D^*(x_h)$	1.466e-2	5.596e-3	2.009e-3	7.162e-4	2.520e-4
$M_R^*(x_h)$	1.262e-1	7.223e-2	3.689e-2	1.880e-2	9.388e-3
I_{eff}	9.53	13.95	19.43	27.29	38.35
$\mathcal{H}^*(x_h, \rho_0)$	0.28531	0.20569	0.14099	0.10160	0.06973
$M_D^*(x_h, \rho_0)$	1.433e-2	5.320e-3	1.855e-3	6.738e-4	2.321e-4
$M_R^*(x_h, \rho_0)$	6.707e-2	3.699e-2	1.802e-2	9.649e-3	4.630e-3
I_{eff}	7.25	10.28	13.89	19.84	27.24
$\mathcal{H}^*(x_h, \hat{\rho}_0)$	0.52454	0.37944	0.26755	0.18779	0.13101
$M_D^*(x_h, \hat{\rho}_0)$	2.443e-2	9.468e-3	3.456e-3	1.234e-3	4.342e-4
$M_R^*(x_h, \hat{\rho}_0)$	2.507e-1	1.345e-1	6.813e-2	3.403e-2	1.673e-3
I_{eff}	13.32	18.97	26.36	36.68	51.18
$\mathcal{H}^*(x_h, \sigma_1)$	0.27382	0.19972	0.13794	0.10007	0.06896
$M_D^*(x_h, \sigma_1)$	1.060e-2	3.972e-3	1.395e-3	5.099e-4	1.764e-4
$M_R^*(x_h, \sigma_1)$	6.437e-2	3.591e-2	1.763e-2	9.503e-3	4.579e-3
I_{eff}	6.95	9.99	13.59	19.54	26.94
$\mathcal{H}^*(x_h, \hat{\sigma}_1)$	0.48249	0.34894	0.24344	0.17108	0.11969
$M_D^*(x_h, \hat{\sigma}_1)$	1.725e-2	6.676e-3	2.413e-3	8.637e-4	3.047e-4
$M_R^*(x_h, \hat{\sigma}_1)$	2.156e-1	1.151e-1	5.685e-2	2.840e-2	1.402e-2
I_{eff}	12.25	17.45	23.98	33.41	46.75
$\mathcal{H}^*(x_h, \rho_1)$	0.19723	0.13034	0.08756	0.06260	0.04279
$M_D^*(x_h, \rho_1)$	7.660e-3	2.595e-3	8.867e-4	3.190e-4	1.095e-4
$M_R^*(x_h, \rho_1)$	3.124e-2	1.439e-2	6.781e-3	3.600e-3	1.721e-3
I_{eff}	5.01	6.52	8.63	12.23	16.71
$\mathcal{H}^*(x_h, \hat{\rho}_1)$	0.99046	1.03801	1.07616	1.09848	1.11085
$M_D^*(x_h, \hat{\rho}_1)$	2.848e-2	1.603e-2	8.617e-3	4.481e-3	2.286e-3
$M_R^*(x_h, \hat{\rho}_1)$	0.95253	1.06144	1.14951	1.20218	1.23170
I_{eff}	25.15	51.90	106.03	214.55	433.93

Figure C.6: Error estimates on uniformly refined meshes, Example B

Level	0	1	2	3	4
$\mid x_0 - x_h \mid_{\Omega,1}$	0.07373	0.03756	0.01892	0.00948	0.00475
$\mathcal{R}_h(x_h)$	0.33072	0.16995	0.08632	0.04347	0.02181
I_{eff}	4.49	4.52	4.56	4.59	4.59
$\mathcal{H}^*(x_h)$	0.40448	0.32759	0.21966	0.14924	0.10274
$M_D^*(x_h)$	2.981e-2	1.223e-2	4.155e-3	1.417e-3	4.880e-4
$M_R^*(x_h)$	1.338e-1	9.508e-2	4.410e-2	2.086e-2	1.007e-2
I_{eff}	5.49	8.72	11.61	15.74	21.63
$\mathcal{H}^*(x_h, \rho_0)$	0.22128	0.11191	0.05626	0.02819	0.01411
$M_D^*(x_h, \rho_0)$	2.037e-2	5.235e-3	1.324e-3	3.324e-4	8.324e-5
$M_R^*(x_h, \rho_0)$	2.860e-2	7.289e-3	1.841e-3	4.623e-4	1.158e-4
I_{eff}	3.00	2.98	2.97	2.97	2.97
$\mathcal{H}^*(x_h, \hat{\rho}_h)$	0.80435	0.57010	0.38279	0.25802	0.17673
$M_D^*(x_h, \hat{\rho}_0)$	7.022e-2	2.592e-2	8.884e-3	3.022e-3	1.039e-3
$M_R^*(x_h, \hat{\rho}_0)$	5.768e-1	2.991e-1	1.376e-1	6.355e-2	3.019e-2
I_{eff}	10.91	15.18	20.23	27.22	37.21
$\mathcal{H}^*(x_h, \sigma_1)$	0.20291	0.10269	0.05164	0.02588	0.01296
$M_D^*(x_h, \sigma_1)$	1.495e-2	3.856e-3	9.770e-4	2.455e-4	6.149e-5
$M_R^*(x_h, \sigma_1)$	2.622e-2	6.688e-3	1.690e-3	4.245e-4	1.064e-4
I_{eff}	2.75	2.73	2.73	2.73	2.73
$\mathcal{H}^*(x_h, \hat{\sigma}_1)$	0.73997	0.53743	0.36211	0.24798	0.17177
$M_D^*(x_h, \hat{\sigma}_1)$	5.148e-2	1.949e-2	6.736e-3	2.333e-3	8.120e-4
$M_R^*(x_h, \hat{\sigma}_1)$	4.961e-1	2.693e-1	1.244e-1	5.916e-2	2.869e-2
I_{eff}	10.04	14.31	19.14	26.16	36.16
$\mathcal{H}^*(x_h, \rho_1)$	0.08948	0.04318	0.02073	0.01015	0.00498
$M_D^*(x_h, \rho_1)$	6.592e-3	1.622e-3	3.922e-4	9.627e-5	2.365e-5
$M_R^*(x_h, \rho_1)$	1.415e-3	2.431e-4	3.760e-5	6.777e-6	1.188e-6
I_{eff}	1.21	1.15	1.10	1.07	1.05
$\mathcal{H}^*(x_h, \hat{\rho}_1)$	1.78613	1.86509	1.92621	1.95939	1.97588
$M_D^*(x_h, \hat{\rho}_1)$	0.10483	0.05665	0.02977	0.01526	0.00772
$M_R^*(x_h, \hat{\rho}_1)$	3.08543	3.42191	3.68050	3.82396	3.89639
I_{eff}	24.23	49.66	101.81	206.69	415.97

Figure C.7: Error estimates on uniformly refined meshes, Example C

Level	0	1	2	3	4
$\mid x_0 - x_h \mid_{\Omega,1}$	0.13255	0.06735	0.03396	0.01703	0.00852
$\mathcal{R}_h(x_h)$	0.55101	0.29426	0.15104	0.07625	0.03827
I_{eff}	4.16	4.37	4.45	4.48	4.49
$\mathcal{H}^*(x_h)$	1.33157	0.74003	0.41710	0.25562	0.16630
$M_D^*(x_h)$	1.803e-1	5.111e-2	1.431e-2	4.368e-3	1.419e-3
$M_R^*(x_h)$	1.59275	0.49654	0.15966	6.098e-2	2.624e-2
I_{eff}	10.05	10.99	12.28	15.01	19.52
$\mathcal{H}^*(x_h, \rho_0)$	0.85229	0.47940	0.24388	0.12237	0.06124
$M_D^*(x_h, \rho_0)$	2.037e-2	5.235e-3	1.324e-3	3.324e-4	8.324e-5
$M_R^*(x_h, \rho_0)$	2.860e-2	7.289e-3	1.841e-3	4.623e-4	1.158e-4
I_{eff}	6.43	7.12	7.18	7.19	7.19
$\mathcal{H}^*(x_h, \hat{\rho}_0)$	1.71703	1.15662	0.72024	0.44639	0.28545
$M_D^*(x_h, \hat{\rho}_0)$	2.601e-1	9.597e-2	3.087e-2	9.674e-3	3.106e-3
$M_R^*(x_h, \hat{\rho}_0)$	2.68809	1.24180	0.48787	0.18959	7.837e-2
I_{eff}	12.95	17.17	21.21	26.21	33.50
$\mathcal{H}^*(x_h, \sigma_1)$	0.81681	0.46074	0.23439	0.11760	0.05885
$M_D^*(x_h, \sigma_1)$	1.064e-1	3.094e-2	7.954e-3	2.002e-3	5.138e-4
$M_R^*(x_h, \sigma_1)$	0.56080	0.18134	4.698e-2	1.183e-2	2.962e-3
I_{eff}	6.16	6.84	6.90	6.91	6.91
$\mathcal{H}^*(x_h, \hat{\sigma}_1)$	1.54737	1.03090	0.64259	0.41593	0.27824
$M_D^*(x_h, \hat{\sigma}_1)$	1.858e-1	6.697e-2	2.150e-2	7.035e-3	2.363e-3
$M_R^*(x_h, \hat{\sigma}_1)$	2.20859	0.99578	0.39142	0.16596	7.505e-2
I_{eff}	11.67	15.31	18.92	24.42	32.66
$\mathcal{H}^*(x_h, \rho_1)$	0.36410	0.12749	0.04866	0.02073	0.00945
$M_D^*(x_h, \rho_1)$	4.419e-2	8.564e-3	1.652e-3	3.529e-4	8.049e-5
$M_R^*(x_h, \rho_1)$	8.838e-2	7.689e-3	7.162e-4	7.681e-5	8.766e-6
I_{eff}	2.75	1.89	1.43	1.22	1.11
$\mathcal{H}^*(x_h, \hat{\rho}_1)$	3.21344	3.51500	3.69285	3.76546	3.79323
$M_D^*(x_h, \hat{\rho}_1)$	0.30363	0.18336	9.979e-2	5.149e-2	2.604e-2
$M_R^*(x_h, \hat{\rho}_1)$	10.0225	12.1719	13.5373	14.1272	14.3626
I_{eff}	24.24	52.19	108.74	221.11	445.21

Figure C.8: Minimising the hypercycle estimates, Example A

Level	0	1	2	3	4
$\mid x_0 - x_h \mid_{\Omega,1}$	0.03938	0.02000	0.01015	0.00512	0.00256
$\mathcal{R}_h(x_h)$	0.16688	0.08781	0.04511	0.02288	0.01151
I_{eff}	4.24	4.39	4.44	4.47	4.50
$\mathcal{H}^*(x_h, \tau_h^{(10)})$	0.28353	0.20671	0.14212	0.10332	0.07198
$M_D^*(x_h, \tau_h^{(10)})$	1.166e-2	4.353e-3	1.483e-3	5.368e-4	1.860e-4
$M_R^*(x_h, \tau_h^{(10)})$	6.873e-2	3.838e-2	1.871e-2	1.014e-2	4.995e-3
I_{eff}	7.20	10.34	14.00	20.18	28.12
$\mathcal{H}^*(x_h, \rho_0^{(10)})$	0.28522	0.20569	0.14099	0.10160	0.06973
$M_D^*(x_h, \rho_0^{(10)})$	1.429e-2	5.318e-3	1.855e-3	6.738e-4	2.321e-4
$M_R^*(x_h, \rho_0^{(10)})$	6.706e-2	3.699e-2	1.802e-2	9.649e-3	4.630e-3
I_{eff}	7.24	10.28	13.89	19.84	27.24
$\mathcal{H}^*(x_h, \hat{\rho}_0^{(10)})$	0.28709	0.20630	0.14123	0.10170	0.06979
$M_D^*(x_h, \hat{\rho}_0^{(10)})$	1.485e-2	5.443e-3	1.880e-3	6.779e-4	2.329e-4
$M_R^*(x_h, \hat{\rho}_0^{(10)})$	6.757e-2	3.712e-2	1.806e-2	9.665e-3	4.638e-3
I_{eff}	7.29	10.32	13.91	19.86	27.26
$\mathcal{H}^*(x_h, \sigma_1^{(10)})$	0.27378	0.19971	0.13794	0.10007	0.06896
$M_D^*(x_h, \sigma_1^{(10)})$	1.059e-2	3.971e-3	1.395e-3	5.099e-4	1.764e-4
$M_R^*(x_h, \sigma_1^{(10)})$	6.437e-2	3.591e-2	1.763e-2	9.503e-3	4.579e-3
I_{eff}	6.95	9.99	13.59	19.54	26.94
$\mathcal{H}^*(x_h, \hat{\sigma}_1^{(10)})$	0.27955	0.20544	0.14018	0.10072	0.06915
$M_D^*(x_h, \hat{\sigma}_1^{(10)})$	1.241e-2	5.257e-3	1.727e-3	5.765e-4	1.889e-4
$M_R^*(x_h, \hat{\sigma}_1^{(10)})$	6.574e-2	3.695e-2	1.792e-2	9.568e-3	4.593e-3
I_{eff}	7.10	10.27	13.81	19.67	27.01
$\mathcal{H}^*(x_h, \rho_1^{(10)})$	0.18356	0.13033	0.08756	0.06260	0.04279
$M_D^*(x_h, \rho_1^{(10)})$	7.120e-3	2.594e-3	8.867e-4	3.192e-4	1.095e-4
$M_R^*(x_h, \rho_1^{(10)})$	2.657e-2	1.439e-2	6.781e-3	3.600e-3	1.721e-3
I_{eff}	4.66	6.52	8.63	12.23	16.71
$\mathcal{H}^*(x_h, \hat{\rho}_1^{(10)})$	0.19931	0.14151	0.09820	0.07114	0.05157
$M_D^*(x_h, \hat{\rho}_1^{(10)})$	9.152e-3	3.313e-3	1.185e-3	4.419e-4	1.594e-4
$M_R^*(x_h, \hat{\rho}_1^{(10)})$	3.057e-2	1.671e-2	8.459e-3	4.620e-3	2.500e-3
I_{eff}	5.06	7.08	9.67	13.89	20.14

Figure C.9: Minimising the hypercycle estimates, Example B

Level	0	1	2	3	4
$\mid x_0 - x_h \mid_{\Omega,1}$	0.07373	0.03756	0.01892	0.00948	0.00475
$\mathcal{R}_h(x_h)$	0.33072	0.16995	0.08632	0.04347	0.02181
I_{eff}	4.49	4.52	4.56	4.59	4.59
$\mathcal{H}^*(x_h, \tau_h^{(10)})$	0.20794	0.10798	0.05593	0.02899	0.01520
$M_D^*(x_h, \tau_h^{(10)})$	1.554e-2	4.105e-3	1.067e-3	2.763e-4	7.235e-5
$M_R^*(x_h, \tau_h^{(10)})$	2.770e-2	7.555e-3	2.061e-3	5.642e-4	1.587e-4
I_{eff}	2.82	2.87	2.96	3.06	3.20
$\mathcal{H}^*(x_h, \rho_0^{(10)})$	0.22120	0.11190	0.05626	0.02819	0.01411
$M_D^*(x_h, \rho_0^{(10)})$	2.033e-2	5.233e-3	1.324e-3	3.324e-4	8.234e-5
$M_R^*(x_h, \rho_0^{(10)})$	2.860e-2	7.289e-3	1.841e-3	4.623e-4	1.158e-4
I_{eff}	3.00	2.98	2.97	2.97	2.97
$\mathcal{H}^*(x_h, \hat{\rho}_0^{(10)})$	0.22450	0.11341	0.05721	0.02942	0.01521
$M_D^*(x_h, \hat{\rho}_0^{(10)})$	2.118e-2	5.406e-3	1.360e-3	3.483e-4	8.991e-5
$M_R^*(x_h, \hat{\rho}_0^{(10)})$	2.922e-2	7.456e-3	1.914e-3	5.173e-4	1.414e-4
I_{eff}	3.04	3.02	3.02	3.10	3.20
$\mathcal{H}^*(x_h, \sigma_1^{(10)})$	0.20288	0.10268	0.05164	0.02588	0.01296
$M_D^*(x_h, \sigma_1^{(10)})$	1.494e-2	3.855e-3	9.769e-4	2.455e-4	6.149e-5
$M_R^*(x_h, \sigma_1^{(10)})$	2.622e-2	6.688e-3	1.690e-3	4.245e-4	1.064e-4
I_{eff}	2.75	2.73	2.73	2.73	2.73
$\mathcal{H}^*(x_h, \hat{\sigma}_1^{(10)})$	0.21197	0.11216	0.05548	0.02754	0.01380
$M_D^*(x_h, \hat{\sigma}_1^{(10)})$	1.748e-2	5.243e-3	1.245e-3	2.890e-4	6.912e-5
$M_R^*(x_h, \hat{\sigma}_1^{(10)})$	2.746e-2	7.337e-3	1.834e-3	4.692e-4	1.212e-4
I_{eff}	2.87	2.99	2.93	2.91	2.91
$\mathcal{H}^*(x_h, \rho_1^{(10)})$	0.08966	0.04321	0.02074	0.01015	0.00498
$M_D^*(x_h, \rho_1^{(10)})$	6.599e-3	1.623e-3	3.923e-4	9.628e-5	2.365e-5
$M_R^*(x_h, \rho_1^{(10)})$	1.439e-3	2.449e-4	3.772e-5	6.786e-6	1.189e-6
I_{eff}	1.22	1.15	1.10	1.07	1.05
$\mathcal{H}^*(x_h, \hat{\rho}_1^{(10)})$	0.15377	0.11267	0.08492	0.06362	0.05257
$M_D^*(x_h, \hat{\rho}_1^{(10)})$	1.317e-2	4.905e-3	1.889e-3	7.206e-4	2.973e-4
$M_R^*(x_h, \hat{\rho}_1^{(10)})$	1.048e-2	7.789e-3	5.322e-3	3.326e-3	2.466e-3
I_{eff}	2.09	3.00	4.49	6.71	11.07

Figure C.10: Minimising the hypercycle estimates, Example C

Level	0	1	2	3	4
$\mid x_0 - x_h \mid_{\Omega,1}$	0.13255	0.06735	0.03396	0.01703	0.00852
$\mathcal{R}_h(x_h)$	0.55101	0.29426	0.15104	0.07625	0.03827
I_{eff}	4.16	4.37	4.45	4.48	4.49
$\mathcal{H}^*(x_h, \tau_h^{(10)})$	0.85154	0.49172	0.25584	0.12998	0.06541
$M_D^*(x_h, \tau_h^{(10)})$	1.201e-1	3.436e-2	8.775e-3	2.222e-3	5.586e-4
$M_R^*(x_h, \tau_h^{(10)})$	6.050e-1	2.074e-1	5.668e-2	1.467e-2	3.720e-3
I_{eff}	6.42	7.30	7.53	7.63	7.68
$\mathcal{H}^*(x_h, \rho_0^{(10)})$	0.85197	0.47937	0.24388	0.12237	0.06124
$M_D^*(x_h, \rho_0^{(10)})$	1.408e-1	4.112e-2	1.059e-2	2.666e-3	6.679e-4
$M_R^*(x_h, \rho_0^{(10)})$	5.850e-1	1.887e-1	4.889e-2	1.231e-2	3.082e-3
I_{eff}	6.43	7.12	7.18	7.19	7.19
$\mathcal{H}^*(x_h, \hat{\rho}_0^{(10)})$	0.85820	0.48161	0.24520	0.12316	0.06173
$M_D^*(x_h, \hat{\rho}_0^{(10)})$	1.465e-1	4.198e-2	1.072e-2	2.691e-3	6.742e-4
$M_R^*(x_h, \hat{\rho}_0^{(10)})$	5.900e-1	1.900e-1	4.940e-2	1.248e-2	3.136e-3
I_{eff}	6.47	7.15	7.22	7.23	7.25
$\mathcal{H}^*(x_h, \sigma_1^{(10)})$	0.81669	0.46073	0.23439	0.11760	0.05885
$M_D^*(x_h, \sigma_1^{(10)})$	1.062e-1	3.093e-2	7.954e-3	2.002e-3	5.014e-4
$M_R^*(x_h, \sigma_1^{(10)})$	5.607e-1	1.813e-1	4.698e-2	1.183e-2	2.962e-3
I_{eff}	6.16	6.84	6.90	6.91	6.91
$\mathcal{H}^*(x_h, \hat{\sigma}_1^{(10)})$	0.83861	0.47780	0.24025	0.11950	0.05951
$M_D^*(x_h, \hat{\sigma}_1^{(10)})$	1.272e-1	4.017e-2	9.509e-3	2.221e-3	5.300e-4
$M_R^*(x_h, \hat{\sigma}_1^{(10)})$	5.760e-1	1.881e-1	4.821e-2	1.206e-2	3.012e-3
I_{eff}	6.33	7.09	7.07	7.02	6.98
$\mathcal{H}^*(x_h, \rho_1^{(10)})$	0.36395	0.12748	0.04866	0.02073	0.00945
$M_D^*(x_h, \rho_1^{(10)})$	4.757e-2	8.561e-3	1.652e-3	3.529e-4	8.049e-5
$M_R^*(x_h, \rho_1^{(10)})$	8.489e-2	7.691e-3	7.163e-4	7.682e-5	8.767e-6
I_{eff}	2.75	1.89	1.43	1.22	1.11
$\mathcal{H}^*(x_h, \hat{\rho}_1^{(10)})$	0.48301	0.26428	0.15529	0.10975	0.09568
$M_D^*(x_h, \hat{\rho}_1^{(10)})$	7.326e-2	2.090e-2	6.275e-3	2.301e-3	9.882e-4
$M_R^*(x_h, \hat{\rho}_1^{(10)})$	1.600e-1	4.894e-2	1.784e-2	9.744e-3	8.166e-3
I_{eff}	3.64	3.92	4.57	6.44	11.23

Figure C.11: Alternative for computing hypercycle estimates, Example A

Level	0	1	2	3	4
$\mathcal{H}^*_{-1}(x_h,\rho_0)$	0.06400	0.03120	0.01507	0.00735	0.00358
$M^*_D(x_h,\rho_0)$	3.214e-3	8.070e-4	1.984e-4	4.871e-5	1.191e-5
$M^*_{-1}(x_h,\rho_0)$	8.829e-4	1.665e-4	2.890e-5	5.251e-6	8.933e-7
I_{eff}	1.63	1.56	1.48	1.44	1.40
$\mathcal{H}^*_{-1}(x_h,\sigma_1)$	0.05251	0.02522	0.01203	0.00581	0.00281
$M^*_D(x_h,\sigma_1)$	2.033e-3	5.016e-4	1.216e-4	2.961e-5	7.182e-6
$M^*_{-1}(x_h,\sigma_1)$	7.244e-4	1.346e-4	2.306e-5	4.153e-6	7.009e-7
I_{eff}	1.33	1.26	1.18	1.14	1.10
$\mathcal{H}^*_{-1}(x_h,\rho_1)$	0.04731	0.02319	0.01129	0.00553	0.00271
$M^*_D(x_h,\rho_1)$	1.838e-3	4.618e-4	1.143e-4	2.820e-5	6.933e-6
$M^*_{-1}(x_h,\rho_1)$	4.007e-4	7.609e-5	1.309e-5	2.389e-6	4.096e-7
I_{eff}	1.20	1.16	1.11	1.08	1.06
$\mathcal{H}^*(x_h,\tilde{\rho}_0)$	0.28534	0.20596	0.19723	0.10183	0.06991
$M^*_D(x_h,\tilde{\rho}_0)$	1.417e-2	5.296e-3	2.009e-3	6.740e-4	2.325e-4
$M^*_R(x_h,\tilde{\rho}_0)$	6.725e-2	3.712e-2	3.689e-2	9.696e-3	4.655e-3
I_{eff}	7.25	10.30	19.42	19.89	27.29
$\mathcal{H}^*(x_h,\tilde{\rho}_0^{(10)})$	0.28517	0.20568	0.14100	0.10162	0.06975
$M^*_D(x_h,\tilde{\rho}_0^{(10)})$	1.426e-2	5.311e-3	1.853e-3	6.735e-4	2.321e-4
$M^*_R(x_h,\tilde{\rho}_0^{(10)})$	6.707e-2	3.700e-2	1.803e-2	9.653e-3	4.632e-3
I_{eff}	7.24	10.28	13.89	19.85	27.23
$\mathcal{H}^*(x_h,\tilde{\sigma}_1)$	0.27380	0.19982	0.13809	0.10018	0.06904
$M^*_D(x_h,\tilde{\sigma}_1)$	1.054e-2	3.959e-3	1.392e-3	5.098e-4	1.765e-4
$M^*_R(x_h,\tilde{\sigma}_1)$	6.442e-2	3.597e-2	1.768e-2	9.526e-3	4.591e-3
I_{eff}	6.95	9.99	13.60	19.57	26.95
$\mathcal{H}^*(x_h,\tilde{\sigma}_1^{(10)})$	0.27378	0.19972	0.13795	0.10007	0.06896
$M^*_D(x_h,\tilde{\sigma}_1^{(10)})$	1.059e-2	3.971e-3	1.395e-3	5.100e-4	1.764e-4
$M^*_R(x_h,\tilde{\sigma}_1^{(10)})$	6.437e-2	3.592e-2	1.763e-2	9.505e-3	4.579e-3
I_{eff}	6.95	9.98	13.59	19.55	26.92
$\mathcal{H}^*(x_h,\tilde{\rho}_1)$	0.18382	0.13085	0.08826	0.06331	0.04359
$M^*_D(x_h,\tilde{\rho}_1)$	7.034e-3	2.579e-3	8.873e-4	3.212e-4	1.111e-4
$M^*_R(x_h,\tilde{\rho}_1)$	2.675e-2	1.454e-2	6.903e-3	3.688e-3	1.789e-3
I_{eff}	4.67	6.54	8.69	12.37	17.02
$\mathcal{H}^*(x_h,\tilde{\rho}_1^{(10)})$	0.18356	0.13044	0.08773	0.06275	0.04290
$M^*_D(x_h,\tilde{\rho}_1^{(10)})$	7.068e-3	2.584e-3	8.854e-4	3.193e-4	1.096e-4
$M^*_R(x_h,\tilde{\rho}_1^{(10)})$	2.663e-2	1.443e-2	6.811e-3	3.619e-3	1.731e-3
I_{eff}	4.66	6.52	8.64	12.26	16.75

Figure C.12: Alternative for computing hypercycle estimates, Example B

Level	0	1	2	3	4
$\mathcal{H}^*_{-1}(x_h, \rho_0)$	0.09986	0.04875	0.02403	0.01192	0.00593
$M^*_D(x_h, \rho_0)$	9.192e-3	2.280e-3	5.654e-4	1.405e-4	3.499e-5
$M^*_{-1}(x_h, \rho_0)$	7.808e-4	9.602e-5	1.189e-5	1.477e-6	1.840e-7
I_{eff}	1.35	1.30	1.27	1.27	1.25
$\mathcal{H}^*_{-1}(x_h, \sigma_1)$	0.08150	0.03952	0.01941	0.00961	0.00478
$M^*_D(x_h, \sigma_1)$	6.005e-3	1.484e-3	3.673e-4	9.112e-5	2.267e-5
$M^*_{-1}(x_h, \sigma_1)$	6.372e-4	7.785e-5	9.603e-6	1.191e-6	1.482e-7
I_{eff}	1.11	1.05	1.03	1.01	1.01
$\mathcal{H}^*_{-1}(x_h, \rho_1)$	0.07464	0.03772	0.01895	0.00949	0.00475
$M^*_D(x_h, \rho_1)$	5.499e-3	1.417e-3	3.585e-4	8.999e-5	2.253e-5
$M^*_{-1}(x_h, \rho_1)$	7.233e-5	6.408e-6	5.183e-7	4.766e-8	4.247e-9
I_{eff}	1.01	1.00	1.00	1.00	1.00
$\mathcal{H}^*(x_h, \tilde{\rho}_0)$	0.22303	0.11454	0.05941	0.03130	0.01689
$M^*_D(x_h, \tilde{\rho}_0)$	2.037e-2	5.332e-3	1.394e-3	3.685e-4	9.955e-5
$M^*_R(x_h, \tilde{\rho}_0)$	2.938e-2	7.787e-3	2.135e-3	6.110e-4	1.856e-4
I_{eff}	3.02	3.05	3.14	3.30	3.56
$\mathcal{H}^*(x_h, \tilde{\rho}_0^{(10)})$	0.22136	0.11212	0.05646	0.02852	0.01447
$M^*_D(x_h, \tilde{\rho}_0^{(10)})$	2.030e-2	5.236e-3	1.328e-3	3.362e-4	8.536e-5
$M^*_R(x_h, \tilde{\rho}_0^{(10)})$	2.869e-2	7.335e-3	1.860e-3	4.775e-4	1.241e-4
I_{eff}	3.00	2.98	2.98	3.01	3.05
$\mathcal{H}^*(x_h, \tilde{\sigma}_1)$	0.20358	0.10424	0.05380	0.02801	0.01480
$M^*_D(x_h, \tilde{\sigma}_1)$	1.494e-2	3.900e-3	1.015e-3	2.653e-4	7.017e-5
$M^*_R(x_h, \tilde{\sigma}_1)$	2.651e-2	6.967e-3	1.880e-3	5.194e-4	1.488e-4
I_{eff}	2.76	2.78	2.84	2.95	3.12
$\mathcal{H}^*(x_h, \tilde{\sigma}_1^{(10)})$	0.20290	0.10273	0.05171	0.02603	0.01315
$M^*_D(x_h, \tilde{\sigma}_1^{(10)})$	1.494e-2	3.857e-3	9.784e-4	2.469e-4	6.240e-5
$M^*_R(x_h, \tilde{\sigma}_1^{(10)})$	2.623e-2	6.696e-3	1.695e-3	4.307e-4	1.105e-4
I_{eff}	2.75	2.73	2.73	2.74	2.77
$\mathcal{H}^*(x_h, \tilde{\rho}_1)$	0.10062	0.05783	0.03621	0.02438	0.01819
$M^*_D(x_h, \tilde{\rho}_1)$	7.340e-3	2.157e-3	6.816e-4	2.304e-4	8.608e-5
$M^*_R(x_h, \tilde{\rho}_1)$	2.784e-3	1.188e-3	6.293e-4	3.640e-4	2.447e-4
I_{eff}	1.36	1.54	1.91	2.57	3.83
$\mathcal{H}^*(x_h, \tilde{\rho}_1^{(10)})$	0.09412	0.04793	0.02705	0.01557	0.00876
$M^*_D(x_h, \tilde{\rho}_1^{(10)})$	6.893e-3	1.795e-3	5.108e-4	1.475e-4	4.156e-5
$M^*_R(x_h, \tilde{\rho}_1^{(10)})$	1.964e-3	5.028e-4	2.209e-4	9.503e-5	3.520e-5
I_{eff}	1.28	1.28	1.43	1.64	1.85

Figure C.13: Alternative for computing hypercycle estimates, Example C

Level	0	1	2	3	4
$\mathcal{H}^*_{-1}(x_h,\rho_0)$	0.20519	0.09705	0.04629	0.02251	0.01109
$M^*_D(x_h,\rho_0)$	3.400e-2	8.328e-3	2.010e-3	4.905e-4	1.209e-4
$M^*_{-1}(x_h,\rho_0)$	8.098e-3	1.091e-3	1.325e-4	1.616e-5	1.991e-6
I_{eff}	1.55	1.44	1.36	1.32	1.30
$\mathcal{H}^*_{-1}(x_h,\sigma_1)$	0.16970	0.07839	0.03680	0.01774	0.00870
$M^*_D(x_h,\sigma_1)$	2.210e-2	5.263e-3	1.249e-3	3.020e-4	7.411e-5
$M^*_{-1}(x_h,\sigma_1)$	6.697e-3	8.815e-4	1.053e-4	1.274e-5	1.563e-6
I_{eff}	1.28	1.16	1.08	1.04	1.02
$\mathcal{H}^*_{-1}(x_h,\rho_1)$	0.14409	0.06887	0.03415	0.01705	0.00852
$M^*_D(x_h,\rho_1)$	1.887e-2	4.627e-3	1.159e-3	2.903e-4	7.261e-5
$M^*_{-1}(x_h,\rho_1)$	1.894e-3	1.171e-4	7.086e-6	4.453e-7	2.787e-8
I_{eff}	1.09	1.02	1.01	1.00	1.00
$\mathcal{H}^*(x_h,\tilde{\rho}_0)$	0.85290	0.48130	0.24589	0.12390	0.06248
$M^*_D(x_h,\tilde{\rho}_0)$	1.395e-1	4.100e-2	1.064e-2	2.696e-3	6.810e-4
$M^*_R(x_h,\tilde{\rho}_0)$	5.879e-1	1.906e-3	4.982e-2	1.266e-2	3.223e-3
I_{eff}	6.43	7.15	7.24	7.28	7.33
$\mathcal{H}^*(x_h,\tilde{\rho}_0^{(10)})$	0.85189	0.47946	0.24415	0.12266	0.06143
$M^*_D(x_h,\tilde{\rho}_0^{(10)})$	1.406e-1	4.104e-2	1.059e-2	2.670e-3	6.697e-4
$M^*_R(x_h,\tilde{\rho}_0^{(10)})$	5.851e-1	1.888e-1	4.902e-2	1.237e-2	3.104e-3
I_{eff}	6.43	7.12	7.19	7.20	7.21
$\mathcal{H}^*(x_h,\tilde{\sigma}_1)$	0.81688	0.46144	0.23539	0.11854	0.05966
$M^*_D(x_h,\tilde{\sigma}_1)$	1.058e-1	3.086e-2	7.969e-3	2.015e-3	5.080e-4
$M^*_R(x_h,\tilde{\sigma}_1)$	5.615e-1	1.821e-1	4.744e-2	1.204e-2	3.052e-3
I_{eff}	6.16	6.85	6.93	6.96	7.00
$\mathcal{H}^*(x_h,\tilde{\sigma}_1^{(10)})$	0.81671	0.46076	0.23444	0.11767	0.05894
$M^*_D(x_h,\tilde{\sigma}_1^{(10)})$	1.063e-1	3.093e-2	7.956e-3	2.003e-3	5.021e-4
$M^*_R(x_h,\tilde{\sigma}_1^{(10)})$	5.608e-1	1.814e-1	4.700e-2	1.185e-2	2.972e-3
I_{eff}	6.16	6.84	6.90	6.91	6.92
$\mathcal{H}^*(x_h,\tilde{\rho}_1)$	0.36896	0.14280	0.07186	0.04439	0.03249
$M^*_D(x_h,\tilde{\rho}_1)$	4.740e-2	9.472e-3	2.421e-3	7.525e-4	2.760e-4
$M^*_R(x_h,\tilde{\rho}_1)$	8.873e-2	1.092e-2	2.742e-3	1.218e-3	7.794e-4
I_{eff}	2.78	2.12	2.12	2.61	3.81
$\mathcal{H}^*(x_h,\tilde{\rho}_1^{(10)})$	0.36545	0.13386	0.05974	0.02986	0.01541
$M^*_D(x_h,\tilde{\rho}_1^{(10)})$	4.732e-2	8.923e-3	2.019e-3	5.074e-4	1.312e-4
$M^*_R(x_h,\tilde{\rho}_1^{(10)})$	8.623e-2	8.996e-3	1.550e-3	3.841e-4	1.062e-4
I_{eff}	2.76	1.99	1.76	1.75	1.81

Figure C.14: On the Impact of Local Mesh Refinement, Example A

	α	Level 1	Level 2	Level 3	Level 4
$\mathcal{H}^*(x_h^{(A)}, \tau_h')$		0.22426	0.17208	0.14064	0.12098
$M_D^*(x_h^{(A)}, \tau_h')$	0.05	1.052e-02	7.909e-03	6.443e-03	5.521e-03
$M_R^*(x_h^{(A)}, \tau_h')$		3.977e-02	2.170e-02	1.334e-02	9.116e-03
$\mathcal{H}^*(x_h^{(A)}, \tau_h')$		0.22410	0.17104	0.13712	0.11140
$M_D^*(x_h^{(A)}, \tau_h')$	0.10	9.922e-03	7.423e-03	5.917e-03	4.793e-03
$M_R^*(x_h^{(A)}, \tau_h')$		4.030e-02	2.183e-02	1.289e-02	7.617e-03
$\mathcal{H}^*(x_h^{(A)}, \tau_h')$		0.22313	0.16739	0.13619	0.11138
$M_D^*(x_h^{(A)}, \tau_h')$	0.15	9.661e-03	7.126e-03	5.770e-03	4.704e-03
$M_R^*(x_h^{(A)}, \tau_h')$		4.012e-02	2.089e-02	1.278e-02	7.701e-03
$\mathcal{H}^*(x_h^{(A)}, \tau_h')$		0.22291	0.16969	0.13589	0.10752
$M_D^*(x_h^{(A)}, \tau_h')$	0.20	9.604e-03	7.104e-03	5.644e-03	4.455e-03
$M_R^*(x_h^{(A)}, \tau_h')$		4.009e-02	2.169e-02	1.282e-02	7.105e-03
$\mathcal{H}^*(x_h^{(A)}, \tau_h')$		0.22255	0.16894	0.13567	0.10680
$M_D^*(x_h^{(A)}, \tau_h')$	0.30	9.506e-03	6.946e-03	5.521e-03	4.328e-03
$M_R^*(x_h^{(A)}, \tau_h')$		4.002e-02	2.160e-02	1.288e-02	7.078e-03
$\mathcal{H}^*(x_h^{(A)}, \tau_h')$		0.22201	0.16799	0.13460	0.10588
$M_D^*(x_h^{(A)}, \tau_h')$	1.00	9.366e-03	6.746e-03	5.334e-03	4.181e-03
$M_R^*(x_h^{(A)}, \tau_h')$		3.992e-02	2.147e-02	1.278e-02	7.031e-03

Figure C.15: On the Impact of Local Mesh Refinement, Example B

	α	Level 1	Level 2	Level 3	Level 4
$\mathcal{H}^*(x_h^{(B)}, \tau_h')$		0.19085	0.16533	0.15237	0.14482
$M_D^*(x_h^{(B)}, \tau_h')$	0.05	1.721e-02	1.450e-02	1.322e-02	1.250e-02
$M_R^*(x_h^{(B)}, \tau_h')$		1.921e-02	1.284e-02	9.992e-03	8.471e-03
$\mathcal{H}^*(x_h^{(B)}, \tau_h')$		0.17381	0.15246	0.13965	0.12664
$M_D^*(x_h^{(B)}, \tau_h')$	0.10	1.536e-02	1.324e-02	1.201e-02	1.076e-02
$M_R^*(x_h^{(B)}, \tau_h')$		1.485e-02	1.001e-02	7.492e-03	5.282e-03
$\mathcal{H}^*(x_h^{(B)}, \tau_h')$		0.16223	0.14263	0.12564	0.11562
$M_D^*(x_h^{(B)}, \tau_h')$	0.15	1.420e-02	1.230e-02	1.065e-02	9.719e-03
$M_R^*(x_h^{(B)}, \tau_h')$		1.212e-02	8.047e-03	5.133e-03	3.650e-03
$\mathcal{H}^*(x_h^{(B)}, \tau_h')$		0.15470	0.13342	0.11737	0.10919
$M_D^*(x_h^{(B)}, \tau_h')$	0.20	1.348e-02	1.139e-02	9.888e-03	9.088e-03
$M_R^*(x_h^{(B)}, \tau_h')$		1.045e-02	6.409e-03	3.887e-03	2.836e-03
$\mathcal{H}^*(x_h^{(B)}, \tau_h')$		0.15184	0.12519	0.11103	0.10042
$M_D^*(x_h^{(B)}, \tau_h')$	0.30	1.313e-02	1.061e-02	9.251e-03	8.259e-03
$M_R^*(x_h^{(B)}, \tau_h')$		9.921e-03	5.060e-03	3.077e-03	1.826e-03
$\mathcal{H}^*(x_h^{(B)}, \tau_h')$		0.14387	0.10774	0.09046	0.08202
$M_D^*(x_h^{(B)}, \tau_h')$	1.00	1.133e-02	8.083e-03	6.699e-03	6.054e-03
$M_R^*(x_h^{(B)}, \tau_h')$		9.371e-03	3.526e-03	1.483e-03	6.734e-04

Figure C.16: On the Impact of Local Mesh Refinement, Example C

	α	Level 1	Level 2	Level 3	Level 4
$\mathcal{H}^*(x_h^{(C)}, \tau_h')$		0.58608	0.48196	0.40951	0.37112
$M_D^*(x_h^{(C)}, \tau_h')$	0.05	9.364e-02	7.554e-02	6.227e-02	5.547e-02
$M_R^*(x_h^{(C)}, \tau_h')$		2.498e-01	1.567e-01	1.054e-01	8.227e-02
$\mathcal{H}^*(x_h^{(C)}, \tau_h')$		0.55343	0.40021	0.34579	0.30247
$M_D^*(x_h^{(C)}, \tau_h')$	0.10	8.319e-02	6.016e-02	5.119e-02	4.335e-02
$M_R^*(x_h^{(C)}, \tau_h')$		2.231e-01	1.000e-01	6.838e-02	4.814e-02
$\mathcal{H}^*(x_h^{(C)}, \tau_h')$		0.54474	0.36366	0.30069	0.26205
$M_D^*(x_h^{(C)}, \tau_h')$	0.15	7.957e-02	5.260e-02	4.294e-02	3.635e-02
$M_R^*(x_h^{(C)}, \tau_h')$		2.172e-01	7.964e-02	4.748e-02	3.232e-02
$\mathcal{H}^*(x_h^{(C)}, \tau_h')$		0.53923	0.34843	0.26826	0.23645
$M_D^*(x_h^{(C)}, \tau_h')$	0.20	7.669e-02	4.838e-02	3.725e-02	3.253e-02
$M_R^*(x_h^{(C)}, \tau_h')$		2.141e-01	7.303e-02	3.472e-02	2.338e-02
$\mathcal{H}^*(x_h^{(C)}, \tau_h')$		0.53699	0.34243	0.24470	0.20267
$M_D^*(x_h^{(C)}, \tau_h')$	0.30	7.610e-02	4.710e-02	3.327e-02	2.742e-02
$M_R^*(x_h^{(C)}, \tau_h')$		2.123e-01	7.016e-02	2.661e-02	1.365e-02
$\mathcal{H}^*(x_h^{(C)}, \tau_h')$		0.53433	0.33509	0.23367	0.18304
$M_D^*(x_h^{(C)}, \tau_h')$	1.00	7.520e-02	4.511e-02	3.110e-02	2.429e-02
$M_R^*(x_h^{(C)}, \tau_h')$		2.103e-01	6.717e-02	2.350e-02	9.212e-03

Figure C.17: Deterioration of I_{eff} due to Incomplete Refinement, Example A

	α	Level 1	Level 2	Level 3	Level 4
$\mathcal{H}^*(\hat{x}_h^{(A)}, \tau_h')$		0.21381	0.15619	0.12310	0.09935
$M_D^*(\hat{x}_h^{(A)}, \tau_h')$	0.05	7.080e-03	4.063e-03	2.941e-03	2.298e-03
$M_R^*(\hat{x}_h^{(A)}, \tau_h')$		3.863e-02	2.033e-02	1.221e-02	7.572e-03
$\mathcal{H}^*(\hat{x}_h^{(A)}, \tau_h')$		0.20898	0.14971	0.11427	0.08630
$M_D^*(\hat{x}_h^{(A)}, \tau_h')$	0.10	6.091e-03	3.150e-03	2.074e-03	1.501e-03
$M_R^*(\hat{x}_h^{(A)}, \tau_h')$		3.758e-02	1.926e-02	1.098e-02	5.947e-03
$\mathcal{H}^*(\hat{x}_h^{(A)}, \tau_h')$		0.20747	0.14859	0.11270	0.08403
$M_D^*(\hat{x}_h^{(A)}, \tau_h')$	0.15	5.735e-03	2.885e-03	1.844e-03	1.262e-03
$M_R^*(\hat{x}_h^{(A)}, \tau_h')$		3.731e-02	1.919e-02	1.086e-02	5.800e-03
$\mathcal{H}^*(\hat{x}_h^{(A)}, \tau_h')$		0.20713	0.14550	0.10929	0.07922
$M_D^*(\hat{x}_h^{(A)}, \tau_h')$	0.20	5.656e-03	2.572e-03	1.545e-03	9.815e-04
$M_R^*(\hat{x}_h^{(A)}, \tau_h')$		3.725e-02	1.860e-02	1.040e-02	5.294e-03
$\mathcal{H}^*(\hat{x}_h^{(A)}, \tau_h')$		0.20655	0.14370	0.10631	0.07557
$M_D^*(\hat{x}_h^{(A)}, \tau_h')$	0.30	5.520e-03	2.281e-03	1.206e-03	6.894e-04
$M_R^*(\hat{x}_h^{(A)}, \tau_h')$		3.714e-02	1.837e-02	1.010e-02	5.022e-03

Figure C.18: Deterioration of I_{eff} due to Incomplete Refinement, Example B

	α	Level 1	Level 2	Level 3	Level 4
$\mathcal{H}^*(\hat{x}_h^{(B)}, \tau_h')$		0.16501	0.12955	0.11076	0.09962
$M_D^*(\hat{x}_h^{(B)}, \tau_h')$	0.05	1.060e-02	6.677e-03	5.252e-03	4.498e-03
$M_R^*(\hat{x}_h^{(B)}, \tau_h')$		1.663e-02	1.011e-02	7.016e-03	5.425e-03
$\mathcal{H}^*(\hat{x}_h^{(B)}, \tau_h')$		0.14707	0.11194	0.09310	0.08054
$M_D^*(\hat{x}_h^{(B)}, \tau_h')$	0.10	9.069e-03	5.596e-03	4.128e-03	3.377e-03
$M_R^*(\hat{x}_h^{(B)}, \tau_h')$		1.256e-02	6.933e-03	4.541e-03	3.110e-03
$\mathcal{H}^*(\hat{x}_h^{(B)}, \tau_h')$		0.13504	0.10132	0.08189	0.06951
$M_D^*(\hat{x}_h^{(B)}, \tau_h')$	0.15	8.159e-03	4.899e-03	3.520e-03	2.818e-03
$M_R^*(\hat{x}_h^{(B)}, \tau_h')$		1.008e-02	5.367e-03	3.187e-03	2.014e-03
$\mathcal{H}^*(\hat{x}_h^{(B)}, \tau_h')$		0.12739	0.09443	0.07501	0.06355
$M_D^*(\hat{x}_h^{(B)}, \tau_h')$	0.20	7.621e-03	4.451e-03	3.185e-03	2.495e-03
$M_R^*(\hat{x}_h^{(B)}, \tau_h')$		8.606e-03	4.465e-03	2.441e-03	1.544e-03
$\mathcal{H}^*(\hat{x}_h^{(B)}, \tau_h')$		0.12427	0.08613	0.06782	0.05493
$M_D^*(\hat{x}_h^{(B)}, \tau_h')$	0.30	7.320e-03	3.964e-03	2.789e-03	2.021e-03
$M_R^*(\hat{x}_h^{(B)}, \tau_h')$		8.122e-03	3.454e-03	1.811e-03	9.962e-04

Figure C.19: Deterioration of I_{eff} due to Incomplete Refinement, Example C

	α	Level 1	Level 2	Level 3	Level 4
$\mathcal{H}^*(\hat{x}_h^{(C)}, \tau_h')$		0.54098	0.42821	0.36998	0.31444
$M_D^*(\hat{x}_h^{(C)}, \tau_h')$	0.05	6.203e-02	4.173e-02	3.328e-02	2.498e-02
$M_R^*(\hat{x}_h^{(C)}, \tau_h')$		2.306e-01	1.416e-01	1.036e-01	7.389e-02
$\mathcal{H}^*(\hat{x}_h^{(C)}, \tau_h')$		0.50392	0.33016	0.28474	0.24707
$M_D^*(\hat{x}_h^{(C)}, \tau_h')$	0.10	5.080e-02	2.650e-02	2.111e-02	1.733e-02
$M_R^*(\hat{x}_h^{(C)}, \tau_h')$		2.031e-01	8.251e-02	5.996e-02	4.371e-02
$\mathcal{H}^*(\hat{x}_h^{(C)}, \tau_h')$		0.49300	0.28749	0.22148	0.19745
$M_D^*(\hat{x}_h^{(C)}, \tau_h')$	0.15	4.651e-02	1.969e-02	1.361e-02	1.171e-02
$M_R^*(\hat{x}_h^{(C)}, \tau_h')$		1.965e-01	6.296e-02	3.544e-02	2.728e-02
$\mathcal{H}^*(\hat{x}_h^{(C)}, \tau_h')$		0.48524	0.26562	0.17725	0.15113
$M_D^*(\hat{x}_h^{(C)}, \tau_h')$	0.20	4.281e-02	1.434e-02	7.795e-03	6.190e-03
$M_R^*(\hat{x}_h^{(C)}, \tau_h')$		1.926e-01	5.622e-02	2.362e-02	1.665e-02
$\mathcal{H}^*(\hat{x}_h^{(C)}, \tau_h')$		0.48267	0.25642	0.14815	0.10783
$M_D^*(\hat{x}_h^{(C)}, \tau_h')$	0.30	4.218e-02	1.296e-02	5.318e-03	3.480e-03
$M_R^*(\hat{x}_h^{(C)}, \tau_h')$		1.908e-01	5.279e-02	1.663e-02	8.148e-03

Figure C.20: The Impact of Local Mesh Refinement on the Obstacle Problem

	α	Level 0	Level 1	Level 2	Level 3	Level 4
$\mathcal{H}^{(\kappa)}(x_h, \tau_h^{(4)})$		0.11889	0.11621	0.11577	0.11536	0.11502
κ	0.05	56.24	71.36	78.12	79.25	78.84
$M_D^{(\kappa)}(x_h, \tau_h^{(4)})$		1.389e-02	1.332e-02	1.323e-02	1.314e-02	1.306e-02
$M_R^{(\kappa)}(x_h, \tau_h^{(4)})$		2.469e-04	1.866e-04	1.694e-04	1.658e-04	1.657e-04
$\mathcal{H}^{(\kappa)}(x_h, \tau_h^{(4)})$		0.11889	0.11561	0.11436	0.11358	0.11324
κ	0.10	56.24	71.58	76.85	76.83	78.04
$M_D^{(\kappa)}(x_h, \tau_h^{(4)})$		1.389e-02	1.318e-02	1.291e-02	1.273e-02	1.266e-02
$M_R^{(\kappa)}(x_h, \tau_h^{(4)})$		2.469e-04	1.841e-04	1.680e-04	1.658e-04	1.623e-04
$\mathcal{H}^{(\kappa)}(x_h, \tau_h^{(4)})$		0.11889	0.11513	0.11293	0.11130	0.11072
κ	0.15	56.24	70.70	74.61	73.32	73.82
$M_D^{(\kappa)}(x_h, \tau_h^{(4)})$		1.389e-02	1.307e-02	1.258e-02	1.222e-02	1.209e-02
$M_R^{(\kappa)}(x_h, \tau_h^{(4)})$		2.469e-04	1.848e-04	1.687e-04	1.667e-04	1.638e-04
$\mathcal{H}^{(\kappa)}(x_h, \tau_h^{(4)})$		0.11889	0.11443	0.11185	0.11021	0.10956
κ	0.20	56.24	69.65	72.57	72.44	72.01
$M_D^{(\kappa)}(x_h, \tau_h^{(4)})$		1.389e-02	1.291e-02	1.234e-02	1.198e-02	1.184e-02
$M_R^{(\kappa)}(x_h, \tau_h^{(4)})$		2.469e-04	1.854e-04	1.701e-04	1.654e-04	1.644e-04
$\mathcal{H}^{(\kappa)}(x_h, \tau_h^{(4)})$		0.11889	0.11342	0.11056	0.10947	0.10906
κ	0.30	56.24	68.34	71.02	71.43	73.27
$M_D^{(\kappa)}(x_h, \tau_h^{(4)})$		1.389e-02	1.268e-02	1.205e-02	1.182e-02	1.173e-02
$M_R^{(\kappa)}(x_h, \tau_h^{(4)})$		2.469e-04	1.855e-04	1.697e-04	1.654e-04	1.601e-04
$\mathcal{H}^*(x_h, \tau_h^{(4)})$		0.12152	0.11826	0.11906	0.12169	0.15440
κ^*	0.05	7.2868	6.4768	4.3286	3.0374	1.1122
$M_D^*(x_h, \tau_h^{(4)})$		4.348e-02	4.098e-02	4.037e-02	4.008e-02	3.955e-02
$M_R^*(x_h, \tau_h^{(4)})$		4.094e-04	4.885e-04	1.077e-03	2.172e-03	1.598e-02
$\mathcal{H}^*(x_h, \tau_h^{(4)})$		0.12152	0.11693	0.11827	0.12311	0.20400
κ^*	0.10	7.2868	6.8143	3.6844	2.3431	0.6583
$M_D^*(x_h, \tau_h^{(4)})$		4.348e-02	4.015e-02	3.908e-02	3.846e-02	3.775e-02
$M_R^*(x_h, \tau_h^{(4)})$		4.094e-04	4.323e-04	1.440e-03	3.503e-03	4.355e-02
$\mathcal{H}^*(x_h, \tau_h^{(4)})$		0.12152	0.11659	0.11895	0.13422	0.25278
κ^*	0.15	7.2868	6.1650	2.9911	1.5153	0.4882
$M_D^*(x_h, \tau_h^{(4)})$		4.348e-02	3.973e-02	3.818e-02	3.765e-02	3.689e-02
$M_R^*(x_h, \tau_h^{(4)})$		4.094e-04	5.227e-04	2.134e-03	8.199e-03	7.740e-02
$\mathcal{H}^*(x_h, \tau_h^{(4)})$		0.12152	0.11597	0.12150	0.15547	0.29962
κ^*	0.20	7.2868	5.9552	2.3874	1.0216	0.3957
$M_D^*(x_h, \tau_h^{(4)})$		4.348e-02	3.924e-02	3.768e-02	3.703e-02	3.645e-02
$M_R^*(x_h, \tau_h^{(4)})$		4.094e-04	5.532e-04	3.305e-03	1.774e-02	1.164e-01
$\mathcal{H}^*(x_h, \tau_h^{(4)})$		0.12152	0.11572	0.12530	0.18087	0.35023
κ^*	0.30	7.2868	5.4753	1.8958	0.7573	0.3244
$M_D^*(x_h, \tau_h^{(4)})$		4.348e-02	3.888e-02	3.685e-02	3.577e-02	3.504e-02
$M_R^*(x_h, \tau_h^{(4)})$		4.094e-04	6.484e-04	5.126e-03	3.119e-02	1.665e-01

Figure C.21: Deterioration of I_{eff} due to Incomplete Refinement

	α	Level 0	Level 1	Level 2	Level 3	Level 4
$\mathcal{H}^{(\kappa)}(x_h, \tau_h^{(10)})$		0.11890	0.06884	0.04745	0.03798	0.03436
κ	0.05	55.95	185.87	323.78	636.57	832.60
$M_D^{(\kappa)}(x_h, \tau_h^{(10)})$		1.389e-02	4.714e-03	2.245e-03	1.440e-03	1.179e-03
$M_R^{(\kappa)}(x_h, \tau_h^{(10)})$		2.482e-04	2.536e-05	6.933e-06	2.262e-06	1.417e-06
$\mathcal{H}^{(\kappa)}(x_h, \tau_h^{(10)})$		0.11890	0.06775	0.04459	0.03040	0.02124
κ	0.10	55.95	181.46	305.67	645.45	1577.54
$M_D^{(\kappa)}(x_h, \tau_h^{(10)})$		1.389e-02	4.566e-03	1.982e-03	9.227e-04	4.510e-04
$M_R^{(\kappa)}(x_h, \tau_h^{(10)})$		2.482e-04	2.516e-05	6.483e-06	1.430e-06	2.859e-07
$\mathcal{H}^{(\kappa)}(x_h, \tau_h^{(10)})$		0.11890	0.06664	0.04200	0.02714	0.01932
κ	0.15	55.95	176.63	324.63	635.05	1889.38
$M_D^{(\kappa)}(x_h, \tau_h^{(10)})$		1.389e-02	4.416e-03	1.759e-03	7.353e-04	3.730e-04
$M_R^{(\kappa)}(x_h, \tau_h^{(10)})$		2.482e-04	2.500e-05	5.418e-06	1.158e-06	1.974e-07
$\mathcal{H}^{(\kappa)}(x_h, \tau_h^{(10)})$		0.11890	0.06565	0.03931	0.02374	0.01707
κ	0.20	55.95	170.02	310.86	533.43	1660.95
$M_D^{(\kappa)}(x_h, \tau_h^{(10)})$		1.389e-02	4.285e-03	1.540e-03	5.626e-04	2.913e-04
$M_R^{(\kappa)}(x_h, \tau_h^{(10)})$		2.482e-04	2.520e-05	4.955e-06	1.055e-06	1.754e-07
$\mathcal{H}^{(\kappa)}(x_h, \tau_h^{(10)})$		0.11889	0.06376	0.03492	0.02081	0.01351
κ	0.30	56.24	165.92	259.97	459.72	969.92
$M_D^{(\kappa)}(x_h, \tau_h^{(10)})$		1.389e-02	4.041e-03	1.215e-03	4.322e-04	1.822e-04
$M_R^{(\kappa)}(x_h, \tau_h^{(10)})$		2.469e-04	2.435e-05	4.673e-06	9.402e-07	1.879e-07
$\mathcal{H}^*(x_h, \tau_h^{(10)})$		0.12152	0.07306	0.05110	0.03939	0.03513
κ^*	0.05	7.2868	6.8694	5.1346	8.7264	10.81
$M_D^*(x_h, \tau_h^{(10)})$		4.348e-02	1.568e-02	7.546e-03	4.595e-03	3.670e-03
$M_R^*(x_h, \tau_h^{(10)})$		4.094e-04	1.662e-04	1.431e-04	3.017e-05	1.570e-05
$\mathcal{H}^*(x_h, \tau_h^{(10)})$		0.12152	0.06880	0.04455	0.03389	0.03073
κ^*	0.10	7.2868	8.5671	8.7435	11.74	8.9962
$M_D^*(x_h, \tau_h^{(10)})$		4.348e-02	1.401e-02	5.878e-03	3.421e-03	2.798e-03
$M_R^*(x_h, \tau_h^{(10)})$		4.094e-04	9.543e-05	3.845e-05	1.241e-05	1.729e-05
$\mathcal{H}^*(x_h, \tau_h^{(10)})$		0.12152	0.06758	0.04197	0.03121	0.02799
κ^*	0.15	7.2868	8.4715	8.1784	11.50	13.28
$M_D^*(x_h, \tau_h^{(10)})$		4.348e-02	1.351e-02	5.208e-03	2.900e-03	2.338e-03
$M_R^*(x_h, \tau_h^{(10)})$		4.094e-04	9.414e-05	3.893e-05	1.097e-05	6.625e-06
$\mathcal{H}^*(x_h, \tau_h^{(10)})$		0.12152	0.06446	0.03997	0.02905	0.02555
κ^*	0.20	7.2868	12.82	7.6755	11.23	14.24
$M_D^*(x_h, \tau_h^{(10)})$		4.348e-02	1.239e-02	4.713e-03	2.511e-03	1.949e-03
$M_R^*(x_h, \tau_h^{(10)})$		4.094e-04	3.771e-05	4.000e-05	9.956e-06	4.804e-06
$\mathcal{H}^*(x_h, \tau_h^{(10)})$		0.12152	0.06283	0.03617	0.02321	0.01547
κ^*	0.30	7.2868	13.32	7.6465	11.67	11.35
$M_D^*(x_h, \tau_h^{(10)})$		4.348e-02	1.178e-02	3.859e-03	1.605e-03	7.124e-04
$M_R^*(x_h, \tau_h^{(10)})$		4.094e-04	3.320e-05	3.300e-05	5.889e-06	2.767e-06

Bibliography

[1] M. Abramowitz and I. A. Stegun (eds.), *Handbook of mathematical functions with formulas, graphs, and mathematical tables. Reprint of the 1972 ed.*, John Wiley & Sons, Inc., New York, 1984.

[2] Robert A. Adams, *Sobolev Spaces*, Academic Press, New York, 1975.

[3] Alfred V. Aho, Ravi Sethi and Jeffrey D. Ullman, *Compilers: Principles, Techniques and Tools*, Addison Wesley, Boston Ma., 1986.

[4] M. Ainsworth and J. T. Oden, *A posteriori error estimators for second order elliptic systems II: An optimal order process for calculating self-equilibrating fluxes*, Comput. Math. Appl. **26** (1993), 75–87.

[5] ———, *A posteriori error estimation in finite element analysis*, Comput. Meth. Appl. Mech. Engrg. **142** (1997), 1–88.

[6] M. Ainsworth, J. T. Oden and C. Y. Lee, *Local a posteriori error estimators for variational inequalities*, Numer. Methods Partial Differ. Equations **9** (1993), 23–33.

[7] R. Araya et al., *A posteriori error estimates for a mixed-FEM formulation of a nonlinear elliptic problem*, Comput. Meth. Appl. Mech. Eng. **191** (2002), 2317 - 2336.

[8] T. Arbogast and Z. Chen, *On the implementation of mixed methods as nonconforming methods for second-order elliptic problems*, Math. Comput. **64** (1995), 943–972.

[9] D. N. Arnold and F. Brezzi, *Mixed and nonconforming finite element methods: Implementation, postprocessing and error estimates*, RAIRO, Model. Math. Anal. Numer. **19** (1985), 7–32.

[10] J. P. Aubin, *Behavior of the error of the approximate solutions of boundary value problems for linear elliptic operators by Galerkin's and finite difference methods*, Ann. Sc. Norm. Super. Pisa **21** (1967), 599–637.

[11] I. Babuška and A. Miller, *The post-processing approach in the finite element method I: Calculation of displacements, stresses and other higher derivatives of the displacements*, Internat. J. Numer. Meth. Engrg. **20** (1984), 1085–1109.

[12] ———, *The post-processing approach in the finite element method II: The calculation of stress intensity factors*, Internat. J. Numer. Meth. Engrg. **20** (1984), 1111–1129.

[13] ———, *The post-processing approach in the finite element method III: A posteriori error estimates and adaptive mesh selection*, Internat. J. Numer. Meth. Engrg. **20** (1984), 2311–2324.

[14] I. Babuška and W. C. Rheinboldt, *A posteriori error estimates for the finite element method*, Internat. J. Numer. Meth. Engrg. **12** (1978), 1597–1615.

[15] ———, *Error estimates for adaptive finite element computations*, SIAM J.Numer. Anal. **15** (1978), 736–754.

[16] ———, *A posteriori error analysis of finite element solutions for one-dimensional problems*, SIAM J. Numer. Anal. **18** (1981), 565 –589.

[17] J. W. Backus, *The syntax and semantics of the proposed international algebraic language of the Zürich ACM-GAMM conference*, Information Processing, Proc. ICIP, UNESCO, Paris 15-20 June 1959, Oldenbourg, München, 1960, pp. 125–132.

[18] Randolph E. Bank and Todd Dupont, *An optimal order process for solving finite element equations*, Math. Comput. **36** (1981), 35–51.

[19] S. Bartels and C. Carstensen, *Each averaging technique yields reliable a posteriori error control in FEM on unstructured grids II: Higher order FEM*, Math. Comput. **71** (2002), 971–994.

[20] R. Becker and R. Rannacher, *A feed-back approach to error control in finite element methods: Basic analysis and examples*, East-West J. Numer. Math. **4** (1996), 237–264.

[21] ———, *An optimal control approach to a posteriori error estimation in finite element methods*, Acta Numerica **10** (2001), 1–102.

[22] J. M. Borwein and Qiji J. Zhu, *A survey of subdifferential calculus with applications*, Nonlinear Analysis, Theory, Methods & Appl. **38A** (1999), 687–773.

[23] H. Blum and F.-T. Suttmeier, *An adaptive finite element discretisation for a simplified Signorini problem*, Calcolo **37** (2000), 65–77.

[24] D. Braess and W. Hackbusch, *A new convergence proof for the multigrid method including the V-cycle*, SIAM J. Numer. Anal. **20** (1983), 967–975.

[25] Dietrich Braess and R. Verfürth, *Multigrid methods for nonconforming finite element methods*, SIAM J. Numer. Anal. **27** (1990), 979–986.

[26] J. H. Bramble and J. E. Pasciak, *New convergence estimates for multigrid algorithms*, Math. Comput. **49** (1987), 311–329.

[27] J. H. Bramble, J. E. Pasciak and A. T. Vassilev, *Analysis of the inexact Uzawa algorithm for saddle point problems*, SIAM J. Numer. Anal. **34** (1997), 1072–1092.

[28] Achi Brandt, *Multi-level adaptive solutions to boundary-value problems*, Math. Comput. **31** (1977), 333–390.

[29] ———, *Multi-level adaptive techniques (MLAT) for partial differential equations: Ideas and software.*, Mathematical Software III, Proc. Symp., Madison 1977 (J. R. Rice, ed.), Academic Press, New York, March 1977, pp. 277–318.

[30] Achi Brandt and Colin W. Cryer, *Multigrid algorithms for the solution of linear complementarity problems arising from free boundary problems*, SIAM J. Sci. Stat. Comput. **4** (1983), 655–684.

[31] S. C. Brenner, *A multigrid algorithm for the lowest-order Raviart-Thomas mixed triangular finite element method*, SIAM J. Numer. Anal. **29** (1992), 647–678.

[32] Haïm Brézis, *Seuil régularité pour certains problèmes unilatéraux*, C. R. Acad. Sci. **273** (1971), 35–37.

[33] H. Brézis and G. Stampacchia, *Sur la régularité de la solution d'inéquations elliptiques*, Bull. Soc. Math. France **96** (1968), 153 –180.

[34] F. Brezzi, J. Douglas and L. D. Marini, *Two families of mixed finite elements for second order elliptic problems*, Numer. Math. **47** (1985), 217–235.

[35] F. Brezzi and M. Fortin, *Mixed and Hybrid Finite Element Methods*, Springer Series in Computational Mathematics, vol. 15, Springer, New York, 1991.

[36] F. Brezzi, W. Hager and P. A. Raviart, *Error estimates for the finite element solution of variational inequalities, Part I, Primal theory*, Numer. Math. **28** (1977), 431–443.

[37] ———, *Error estimates for the finite element solution of variational inequalities, Part II, Mixed methods*, Numer. Math. **31** (1978), 1–16.

[38] C. Bron, *Algorithm 426: Merge Sort Algorithm [M1]*, Comm. ACM (1972), 357–358.

[39] Hinderk M. Buß, *On the Implementation of Multilevel Solvers for Constrained Variational Problems in Dual Formulation*, Fast Solution of Discretized Optimization Problems (ISNM 138) (K.-H. Hoffmann, ed.), Birkhäuser, Weierstrass Institute for Applied Analysis and Stochastics, Basel, 2000, pp. 58–72.

[40] ———, *Finite Element Methods for Variational Problems Based on Nonconforming Dual Mixed Discretisations*, East West J. Numer. Math. **9** (2001), 77–98.

[41] Hinderk M. Buß and Sergey I. Repin, *A posteriori Error Computation for an Obstacle Problem*, Technical Report 55, Sonderforschungsbereich 359, University Heidelberg, 1999.

[42] ———, *A posteriori Error Estimates for Boundary Value Problems with Obstacles*, ENUMATH 99. Numerical Mathematics and Advanced Applications, Proc. 3rd Conf. Jyväskylä 1999 (P. Neittaanmäki et al., eds.), World Scientific, Singapore, 2000, pp. 162–170.

[43] Zhiqiang Cai, Charles I. Goldstein and Joseph E. Pasciak, *Multilevel iteration for mixed finite element systems with penalty*, SIAM J. Sci. Comput. **14** (1993), 1072–1088.

[44] C. Carstensen and S. Bartels, *Each averaging technique yields reliable a posteriori error control in FEM on unstructured grids I: Low order conforming, nonconforming, and mixed FEM*, Math. Comput. **71** (2002), 945–969.

[45] Zhiming Chen and Ricardo H. Nochetto, *Residual type a posteriori error estimates for elliptic obstacle problems*, Numer. Math. **84** (2000), 527–548.

[46] Xiao Liang Cheng, *On the nonlinear inexact Uzawa algorithm for saddle-point problems*, SIAM J. Numer. Anal. **37** (2000), 1930–1934.

[47] F. Christian and G. Santos, *A posteriori estimators for nonlinear elliptic partial differential equations*, J. Comput. Appl. Math. **103** (1999), 447–459.

[48] P. G. Ciarlet, *The Finite Element Method for Elliptic Problems*, Studies in Mathematics and its Applications, vol. 4, Elsevier North-Holland Inc., New York, 1978.

[49] P. G. Ciarlet and P. A. Raviart, *General Lagrange and Hermite interpolation in $\mathbb{R}^n$ with applications to finite element methods*, Arch. rat. Mech. Anal. **46** (1972), 177–199.

[50] P. G. Ciarlet and C. Wagschal, *Multipoint Taylor formulas and applications to the finite element method*, Numer. Math. **17** (1971), 84–100.

[51] Frank H. Clarke, *Generalized gradients and applications*, Trans. Am. Math. Soc. **205** (1975), 247–262.

[52] Ph. Clément, *Approximation by finite element functions using local regularization*, RAIRO Anal. Numer. R-2 **9** (1975), 77–84.

[53] Richard Courant, *Variational methods for the solution of problems of equilibrium and vibrations*, Bull. Am. Math. Soc. **49** (1943), 1–23.

[54] M. Crouzeix and P. A. Raviart, *Conforming and nonconforming finite element methods for solving the stationary Stokes equations I*, Revue Franc. Automat. Inform. Rech. operat. R-3 **7** (1973), 33–76.

[55] Colin W. Cryer, *The solution of a quadratic programming problem using systematic overrelaxation*, SIAM J. Control **9** (1971), 385–392.

[56] D. A. D'Esopo, *A convex programming procedure*, Naval Res. Logist. Quart. **6** (1959).

[57] Gianni Dal Maso and Igor V. Skrypnik, *Capacity theory for monotone operators*, Potential Anal. **7** (1997), 765–803.

[58] J. Douglas and J. E. Roberts, *Mixed finite element methods for second order elliptic problems*, Mat. Apl. Comput. **1** (1982), 91–103.

[59] T Dupont and R. Scott, *Polynomial approximation of functions in Sobolev spaces*, Math. Comput. **34** (1980), 441–463.

[60] I. Ekeland and R. Temam, *Convex analysis and variational problems*, Studies in Mathematics and its Applications, vol. 1, North-Holland Publishing Company, Amsterdam-Oxford, 1976.

[61] K. Eriksson, D. Estep, P. Hansbo and C. Johnson, *Introduction to adaptive methods for differential equations*, Acta Numerica 1995 (A. Iserles, ed.), Cambridge University Press, Cambridge, 1995, pp. 105–158.

[62] K. Eriksson and C. Johnson, *An adaptive finite element method for linear elliptic problems*, Math. Comput. **50** (1988), 361–383.

[63] ———, *Adaptive finite element methods for parabolic problems I: A linear model problem*, SIAM J. Numer. Anal. **28** (1991), 43–77.

[64] Lawrence C. Evans and Ronald F. Gariepy, *Measure theory and fine properties of functions*, Studies in Advanced Mathematics, CRC Press, Boca Raton, 1992.

[65] R. S. Falk, *Error estimates for the approximation of a class of variational inequalities*, Math. Comp. **28** (1974), 963–971.

[66] R.P. Fedorenko, *A relaxation method for solving elliptic difference equations*, U.S.S.R. Comput. Math. Math. Phys. **1962** (1961), 1092–1096. (Russian, English)

[67] W. Fenchel, *On conjugate convex functions*, Can. J. Math. **1** (1949), 73–77.

[68] ———, *Convex cones, sets and functions*, Logistic Project Report, Department of Mathematics, Princeton University, Princeton, 1953.

[69] G. Fichera, *Problemi elastostatici con vincoli unilaterali: Il problema di Signorini con ambigue condizioni al contorno*, Atti Accad. Naz. Lincei, Mem., Cl. Sci. Fis. Mat. Nat., Sez. I, VIII. Ser. **7** (1964), 91–140.

[70] G. J. Fix, *On the effects of quadrature in the finite element method*, Advances in Computational Methods in Structural Mechanics and Design (J. T. Oden, R. W. Clough, and Y. Yamamoto, eds.), The University of Alabama Press, Huntsville, 1972, pp. 55–68.

[71] B. M. Fraeijs de Veubeke, *Displacement and Equilibrium Models in the Finite Element Method*, Sress Analysis (O. C. Zienkiewicz and G. S. Holister, eds.), Wiley, London, 1965, pp. 145–197.

[72] Jens Frehse, *On Signorini's problem and variational problems with thin obstacles*, Ann. Sc. Norm. Super. Pisa, Cl. Sci., IV. Ser. **4** (1977), 343–362.

[73] Donald A. French, Stig Larsson and Ricardo H. Nochetto, *Pointwise a posteriori error analysis for an adaptive penalty finite element method for the obstacle problem*, Comput. Methods Appl. Math. **1** (2001), 18–38.

[74] Avner Friedman, *Variational principles and free-boundary problems*, Wiley, New York, 1982.

[75] A. George and J. W. Liu, *Computer Solution of Large Sparse Positive Definite Systems*, Prentice-Hall Series in Computational Mathematics, Prentice-Hall Inc., Englewood Cliffs, New Jersey, 1981.

[76] V. Girault and P. A. Raviart, *Finite Element Methods for Navier-Stokes Equations*, Springer Series in Computational Mathematics 5, Springer, Berlin, 1986.

[77] Dick Grune and Ceriel J. H. Jacobs, *Parsing Techniques - A Practical Guide*, Ellis Horwood Ltd., Chichester, 1990.

[78] W. Hackbusch and H. D. Mittelmann, *On multi-grid methods for variational inequalities*, Numer. Math. **42** (1983), 65–76.

[79] R. J. Herbold, *Consistent Quadrature Schemes for the Numerical Solution of Boundary Value Problems by Variational Techniques*, Ph.D. Thesis, Case Western Reserve University, Cleveland, 1968.

[80] R. J. Herbold, M. H. Schultz and R. S. Varga, *The effect of quadrature errors in the numerical solution of boundary value problems by variational techniques*, Aequationes Math. **3** (1969), 247–270.

[81] Clifford Hildreth, *A quadratic programming procedure*, Naval Res. Logist. Quart. **4** (1957).

[82] I. Hlaváček, J. Haslinger, J. Nečas and J. Lovíšek, *Solution of variational inequalities in mechanics*, Applied Mathematical Sciences, vol. 66, Springer, New York, 1988.

[83] I. Hlaváček, M. Křížek and V. Pistora, *How to recover the gradient of linear elements on nonuniform triangulations.*, Appl. Math., Praha **41** (1996), 241–267.

[84] C. A. R. Hoare, *Quicksort*, Comput. J. **5** (1962), 10–15.

[85] R. H. W. Hoppe, *Two-sided approximations for unilateral variational inequalities by multi-grid methods*, Optimization **18** (1987), 867–881.

[86] Claes Johnson, *Adaptive finite element methods for the obstacle problem*, Math. Models Methods Appl. Sci. **2** (1992), 483–487.

[87] Jim E. Jones and Panayot S. Vassilevski, *AMGe based on Element Agglomeration*, SIAM J. Sci. Comput. **23** (2001), 109–133.

[88] David Kinderlehrer, *The smoothness of the solution of the boundary obstacle problem*, J. Math. Pures Appl., IX. Sér. **60** (1981), 193–212.

[89] D. Kinderlehrer and G. Stampacchia, *An Introduction to Variational Inequalities and their Applications*, Pure and Applied Mathematics, vol. 88, Academic Press, New York, 1980.

[90] Donald E. Knuth, *The Art of Computer Programming Vol. 3: Sorting and Searching*, Addison-Wesley Series in Computer Science and Information Processing, Addison-Wesley, Reading Mass., 1974.

[91] Ralf Kornhuber, *Monotone multigrid methods for elliptic variational inequalities I*, Numer. Math. **69** (1994), 167–184.

[92] ———, *Monotone multigrid methods for elliptic variational inequalities II*, Numer. Math. **72** (1996), 481–499.

[93] ———, *A Posteriori Error Estimates for Elliptic Variational Inequalities*, Comput. Math. Applic. **31** (1996), 49–60.

[94] M. Křížek and P. Neittaanmäki, *On a global superconvergence of the gradient of linear triangular elements*, J. Comput. Appl. Math. **18** (1987), 221–233.

[95] ———, *On superconvergence techniques*, Acta Appl. Math. **9** (1987), 175–198.

[96] ———, *Bibliography on superconvergence*, Finite Element Methods. Superconvergence, Post-Processing, and a Posteriori Estimates (M. Křížk et al., eds.), Marcel Dekker Inc., New York, 1998, pp. 315–348, Lect. Notes Pure Appl. Math. 196.

[97] A. Kufner, O. John and S. Fučík, *Function Spaces*, Noordhoff International Publishing, Leyden, 1977.

[98] P. Ladevèze and D. Leguillon, *Error estimate procedure in the finite element method and applications*, SIAM J. Numer. Anal. **20** (1983), 485–509.

[99] P. D. Lax and A. N. Milgram, *Parabolic equations*, Ann. Math. Studies **33** (1954), 167–190.

[100] J. Levine, T. Mason and D. Brown, *Lex & Yacc*, 2nd ed., O'Reilly, Cambridge Ma., 1992.

[101] Stampacchia 1969 Lewy, H. Lewy and G. Stampacchia, *On the regularity of the solution of a variational inequality*, Comm. Pure. Appl. Math. **22** (1969), 153–188.

[102] J. L. Lions and G. Stampacchia, *Variational inequalities*, Commun. Pure Appl. Math. **20** (1967), 493–519.

[103] Wenbin Liu and Ningning Yan, *A Posteriori Error Estimates for a Class of Variational Inequalities*, J. Sci. Comput. **15** (2000), 361–393.

[104] Y. Maday, A. Patera and J. Peraire, *A general formulation for a posteriori bounds for output functionals of partial differential equations; application to the eigenvalue problem*, C. R. Acad. Sci., Paris, Sér. I, Math. **328** (1999), 823 –828.

[105] N. L. Majorova, *On the existence of a minimum of a functional following from the S-property of its quasi-differential*, Qualitative Methods for Investigating Operator Equations (V. S. Klimov, ed.), Yaroslavskij Gosudarstvennyj Universitet, Yaroslavl, 1988, pp. 55–62.

[106] Jan Mandel, *A multilevel iterative method for symmetric, positive definite linear complementarity problems*, Appl. Math. Optimization **11** (1984), 77–95.

[107] L. D. Marini, *An inexpensive method for the evaluation of the solution of the lowest order Raviart-Thomas mixed method*, SIAM J. Numer. Anal. **22** (1985), 493–496.

[108] Vladimir G. Maz'ya, *Sobolev Spaces*, Springer, Berlin, 1985, Transl. from the Russian by T. O. Shaposhnikova.

[109] S. G. Mikhlin, *Variational Methods in Mathematical Physics*, International Series of Monographs in Pure and Applied Mathematics, vol. 50, Pergamon, Oxford, 1964.

[110] Peter Naur et al. (eds.), *Report on the algorithmic language Algol 60*, Numer. Math. **2** (1960), 106–136.

[111] Tilman Neunhöffer, *Multigrid Methods for Mixed Finite Element Discretizations of Variational Inequalities*, Multigrid Methods IV (P. W. Hemker, ed.), Birkhäuser, ISNM 116, Basel, 1994, pp. 257–268.

[112] R. A. Nicolaides, *Existence, uniqueness and approximation for generalized saddle point problems*, SIAM J. Numer. Anal. **19** (1982), 349–357.

[113] Joachim A. Nitsche, *Ein Kriterium für die Quasi-Optimalität des Ritzschen Verfahrens*, Numer. Math. **11** (1968), 346–348.

[114] Ricardo H. Nochetto, *Sharp L^∞-Error Estimates for Semilinear Elliptic Problems with Free Boundaries*, Numer. Math. **54** (1988), 243–255.

[115] M. Paraschivoiu et al., *Fast bounds for outputs of parallel differential equations*, Computational Methods for Optimal Design and Control, Proc. 2nd AFOSR Workshop on Optimal Design and Control 1997 (J. Borggaard et al., eds.), Prog. Syst. Control Theory, vol. 24, Birkhäuser, Boston, 1998, pp. 323–360.

[116] J. Peraire and A. Patera, *Bounds for Linear-Functional Outputs of Coercive Partial Differential Equations: Local Indicators and Adaptive Refinement*, Advances in Adaptive Computational Methods in Mechanics (P. Ladevéze and J. Oden, eds.), Studies in Applied Mechanics, vol. 47, Elsevier, Amsterdam, 1998, pp. 199–216.

[117] J. Pousin and J. Rappaz, *Consistency, stability, a priori and a posteriori errors for Petrov-Galerkin methods applied to nonlinear problems*, Numer. Math. **69** (1994), 213–231.

[118] W. Prager and J. L. Synge, *Approximations in elasticity based on the concept of function spaces*, Quart. Appl. Math. **5** (1947), 241 –269.

[119] Rolf Rannacher, *A posteriori error estimation in least-squares stabilized finite element schemes*, Comput. Methods Appl. Mech. Eng. **166** (1998), 99–114.

[120] R. Rannacher, *Adaptive Galerkin finite element methods for partial differential equations*, J. Comput. Appl. Math. **128** (2001), 205–233.

[121] R. Rannacher and F.-T. Suttmeier, *A posteriori error control in finite element methods via duality techniques: Application to perfect plasticity*, Comput. Mech. **21** (1998), 123–133.

[122] P. A. Raviart and J. M. Thomas, *A mixed finite element method for 2-nd order elliptic problems*, Math. Aspects Finite Elem. Meth., Proc. Conf. Rome 1975 (I. Galligani and E. Magenes, eds.), Lect. Notes Math., vol. 606, Springer, Berlin, 1977, pp. 292–315.

[123] Sergey I. Repin, *A posteriori error estimation for nonlinear variational problems by duality theory*, Zapiski Nauchnych Seminarov V. A. Steklov Mathematical Institute (POMI) **243** (1997), 201–214.

[124] ———, *A Posteriori Error Estimates for Approximate Solutions of Variational Problems with Power Growth Functionals*, Zapiski Nauchnych Seminarov V. A. Steklov Mathematical Institute (POMI) **249** (1997), 244–255.

[125] ———, *A posteriori error estimation for variational problems with uniformly convex functionals*, Math. Comp. **69** (2000), 481–500.

[126] Sergey I. Repin and L. S. Xanthis, *A posteriori error estimation for elasto-plastic problems based on duality theory*, Comput. Methods Appl. Mech. Engrg. **138** (1996), 317–339.

[127] R. T. Rockafellar, *Duality and stability in extremum problems involving convex functions*, Pac. J. Math. **21** (1967), 167–187.

[128] ———, *Conjugate Duality and Optimisation*, CBMS Regional Conference Series in Applied Mathematics, vol. 16, SIAM, Philadelphia, Pa., 1974.

[129] H. Samelson, O. Wesler and R. M. Thrall, *A partition theorem for Euclidean n-space*, Proc. Amer. Math. Soc. **9** (1958), 805–807.

[130] Russell Schaffer and Robert Sedgewick, *The analysis of heapsort*, J. Algorithms **15** (1993), 76–100.

[131] Rainer Schumann, *Regularity for Signorini's problem in linear elasticity*, Manuscr. Math. **63** (1989), 255–291.

[132] Jonathan Richard Shewchuk, *Delaunay Refinement Algorithms for Triangular Mesh Generation*, Computational Geometry: Theory and Applications **22** (2002), 21–74.

[133] Peter Sonneveld, *CGS, a fast Lanczos-type solver for nonsymmetric linear systems*, SIAM J. Sci. Stat. Comput. **10** (1989), 36–52.

[134] S. P. Vanka, *Block-implicit multigrid solution of Navier-Stokes equations in primitive variables*, J. Comput. Phys. **65** (1986), 138–158.

[135] Waldemar Velte, *Direkte Methoden der Variationsrechnung*, Teubner Studienbücher, Teubner, Stuttgart, 1976.

[136] Rüdiger Verfürth, *A Posteriori Error Estimates for Nonlinear Problems. Finite Element Discretizations of Elliptic Equations*, Math. Comp. **62** (April 1994), 445–475.

[137] ———, *A Review of A Posteriori Error Estimation and Adaptive Mesh-Refinement Techniques*, Advances in Numerical Mathematics, Wiley & Sons, Chichester, 1995.

[138] Rüdiger. Verfürth, *A Review of A Posteriori Error Estimation and Adaptive Mesh-Refinement Techniques*, Advances in Numerical Mathematics, Wiley & Teubner, Stuttgart, 1996.

[139] A. A. Vladimirov, Yu. E. Nesterov and Yu. N. Chekanov, *On uniformly convex functionals*, Mosc. Univ. Comput. Math. Cybern. **3** (1979), 10–21.

[140] M. A. Wolfe, *Numerical Methods for Unconstrained Optimization. An Introduction*, Van Nostrand Reinhold Company VIII, New York, 1978.

[141] C. Zalinescu, *On uniformly convex functions*, J. Math. Anal. Appl. **95** (1983), 344–374.

[142] O. C. Zienkiewicz and J. Z. Zhu, *A simple error estimator and adaptive procedure for practical engineering analysis*, Int. J. Numer. Meth. Engrg. **24** (1987), 337–357.

List of Figures